

T_EX: tabela ważniejszych komend

HASŁO	KOMENDA	PRZYKŁAD I KOMENTARZ	DOMYŚLNIE
krój pisma	<code>\it</code> <code>\sl</code> <code>\bf</code> <code>\tt</code> <code>\rm</code>	<code>{\it kursywa}</code> — <i>kursywa</i> <code>{\sl pochyły}</code> — <i>pochyły</i> <code>{\bf pogrubiony}</code> — pogrubiony <code>{\tt maszynowy}</code> — maszynowy <code>{\rm antykwa}</code> — antykwa (pismo proste)	10-punktowa antykwa (<code>\rm</code>)
wyśrodkowanie	<code>\centerline</code>	<code>\centerline{tekst}</code> wyśrodkowuje tekst w wierszu UWAGA: Aby zastosować tę komendę w środku akapitu, należy najpierw złamać wiersz (patrz: „podział wierszy”)	
odstęp między wierszami	<code>\baselineskip</code>	<code>\baselineskip 9 pt</code> tworzy 9-punktowy odstęp między wierszami	12 pt
podział wierszy	<code>\hfil\break</code>	kończy bieżący wiersz	
inne odstępy	<code>\smallskip</code> <code>\medskip</code> <code>\bigskip</code> <code>\quad, \qquad</code>	mały odstęp pionowy średni odstęp pionowy duży odstęp pionowy odstępy poziome	
akapity	<code>\parindent</code> <code>\parskip</code> <code>\noindent</code>	<code>\parindent 1 in</code> ustawia 1-calowe wcięcie akapitu <code>\parskip 1 cm</code> wstawia 1 cm odstępu między akapitami <code>\noindent</code> lokalnie zakazuje wcięcia akapitu	20 pt 0 pt (elastyczne)
matematyka	<code>\$ wzór \$</code> <code>\$\$ wzór \$\$</code>	wzory w tekście wzory eksponowane	
powiększanie dokumentu	<code>\magnification</code> <code>\magstep...</code>	<code>\magnification 1200</code> daje 20-procentowe powiększenie <code>\magnification=\magstep1</code> działa tak samo UWAGA: ta komenda może być wykorzystana wyłącznie na początku pliku	1000
rozmiar kolumny	<code>\hsize</code> <code>\vsize</code>	<code>\hsize 3 true cm</code> — 3-centymetrowa szerokość <code>\vsize 8 true in</code> — 8-calowa wysokość <code>\vsize 8 in</code> — 8-calowa wysokość, powiększana po umieszczeniu komendy <code>\magnification</code>	6,5 true in 8,9 true in
położenie kolumny na stronie	<code>\hoffset</code> <code>\voffset</code>	<code>\hoffset 1 true in</code> przesuwa kolumnę w prawo o 1 cal <code>\hoffset -1 true in</code> przesuwa kolumnę w lewo o 1 cal <code>\voffset 1 true in</code> przesuwa kolumnę w dół o 1 cal <code>\voffset -1 true in</code> przesuwa kolumnę w górę o 1 cal	0 in 0 in
numeracja stron	<code>\pageno</code> <code>\nopagenumbers</code>	<code>\pageno=9</code> rozpoczyna numerację stron od 9 <code>\pageno=-9</code> tak samo, tylko cyframi rzymskimi wyłącza automatyczną numerację stron	<code>\pageno=1</code>
koniec pracy	<code>\end</code> (lub <code>\bye</code>)	komenda, która musi znaleźć się na końcu pliku	

Tabela ta jest zestawieniem podstawowych konwencji T_EX-a; inne pakiety kierują się zupełnie innymi regułami. Komendy tworzące znaki specjalne czy skomplikowane formaty, jak np. tabele, są wymienione w dodatku.

T_EX w przykładach

Podręcznik dla początkujących

T_EX w przykładach

Podręcznik dla początkujących

Arvind Borde

ACADEMIC PRESS, INC.
Harcourt Brace Jovanovich, Publishers

Boston SanDiego New York
London Sydney Tokyo Toronto

This book is printed on acid-free paper. (∞)

Copyright © 1992 by Academic Press, Inc.
All rights reserved.

No part of this publication may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopy, recording, or any information storage and retrieval system, without permission in writing from the publisher.

ACADEMIC PRESS, INC.
1250 Sixth Avenue, San Diego, CA 92101

United Kingdom Edition published by
ACADEMIC PRESS LIMITED
24-28 Oval Road, London NW1 7DX

Library of Congress Cataloging-in-Publication Data

Printed in the United States of America
91 92 93 94 9 8 7 6 5 4 3 2 1

This book is a significantly expanded version of author's earlier introduction to T_EX, *An Absolute Beginner's Guide to Using T_EX* (© Arvind Borde, 1987).

Apart from its covers, the book was typeset entirely in Plain T_EX by the author, with assistance from the production department of Academic Press, Boston. The covers were done partially in T_EX. The final copy was produced by the American Mathematical Society on an Autologic ASP Micro-5 phototypesetter.

'T_EX' is a trademark of the American Mathematical Society.

Moim rodzicom

```

1 \hrule
2 \vfil
3 \centerline{\vrule height ,06 in width 5 in}
4 \bigskip
5 \centerline{\bf JAK KORZYSTAĆ Z PODRĘCZNIKA}
6 \bigskip
7 {\narrower\narrower\narrower\narrower
8 \noindent Przewodnik ten składa się z dwóch przeplatających się nawzajem części.
9 Na stronach nieparzystych (po prawej), zwanych stronami wyjściowymi ({\it output
10 pages\}), przedstawiono podstawowe zastosowania programu; a dokładniej ---
11 zilustrowano te „rzeczy”, które można zrobić za pomocą TeX-a. Na stronach
12 parzystych (po lewej) znajdują się pudełka ({\it boxes}), które pokazują, co
13 napisano, aby TeX złożył stronę wyjściową (wynikową). Patrząc na pudełka
14 wejściowe ({\it input boxes\}), można od razu zobaczyć osiągnięty efekt
15 składu, np. {\bf pogrubiony} krój pisma, czy symbole $\int$ lub $\heartsuit$,
16 bez konieczności czytania długich wyjaśnień. Linijki tekstu w pliku wejściowym
17 są dla wygody ponumerowane, a w miejscach wymagających wyjaśnienia dodano
18 kilka uwag.
19
20 Przykłady zawarte w tej książce obejmują wiele standardowych zastosowań TeX-a.
21 Można korzystać z niej „wizualnie” --- przeglądać kolejne strony do momentu,
22 kiedy natrafi się na interesujące pojęcie, i wtedy szukać jego odpowiednika
23 w pliku wejściowym (na stronie parzystej). Jeśli napotka się na problemy
24 z odnalezieniem owego zagadnienia (ponieważ niemożliwe jest przedstawienie
25 i omówienie wszystkiego, co {\it może\} się wydarzyć), należy szukać go
26 w dodatku. Znajduje się tam alfabetyczny spis wielu zastosowań programu
27 i większości powszechnie wykorzystywanych komend.\bigskip}
28 \centerline{\vrule height ,06 in width 5 in}
29 \vfil\vfil
30 \hrule
31 \pageno=-7
32 \eject
33 \end

```

UWAGI

ROZPOCZYNAJMY PRACĘ Z T_EX-EM

Tekst wydrukowany na stronie obok powstał na podstawie instrukcji umieszczonych w ramce powyżej. Zawartość tej ramki to „zwykły tekst” (znaki ASCII), tak samo jak i komendy. Komendy zazwyczaj poprzedzone są znakiem \.

Na stronie obok znajduje się kolumna tekstu o domyślnie ustawionym rozmiarze (6,5 cala na 8,9 cala, z 1-calowym marginesem u góry i po lewej). Rozmiar kolumny i jej położenie na stronie określają cienkie kreski poziome u góry i u dołu strony składu. Większość tekstu złożono domyślnym krojem pisma.

1, 30: `\hrule` składa poziomą kreskę.

2, 29: `\vfil` powoduje umieszczenie w obrębie kolumny tekstu pionowego odstępu. Komendy znajdujące się w tych wierszach utrzymują w równowadze blok tekstu, z odstępem dwa razy większym na dole (dwie komendy `\vfil`) niż u góry kolumny.

3, 5, 28: `\centerline{zawartość}` wyśrodkowuje „zawartość” w wierszu. W wierszach 3 i 27 jest to `\vrule`, pionowa kreska o ustalonej (w *calach*) wysokości i szerokości. Komenda ta składa grube kreski nad i pod tekstem.

4, 6, 27: `\bigskip` wstawia duży odstęp pionowy.

5, 15: `\bf` składa tekst pogrubioną odmianą kroju pisma. Tekst występujący w tej odmianie musi być ograniczony nawiasami `{ i }` (chyba że pozostała część dokumentu ma być złożona takim krojem).

7: Każda komenda `\narrower` powoduje zmniejsze-

nie szerokości tekstu akapitu (wcięcie z dwóch stron). Komendy te odnoszą się do wierszy 7–27: nawiasy grupujące `{ i }` określają, jaka część tekstu ma być złożona w taki sposób.

8: `\noindent` likwiduje wcięcie danego akapitu.

9, 12, 14: `\it` składa tekst kursywą.

11, 20: `\TeX` składa logo „T_EX”.

15: Pewne symbole są dostępne jedynie w „matematycznym trybie pracy”. Znak `$` uaktywnia ten tryb (kiedy pojawia się po raz pierwszy) i informuje o jego opuszczeniu (gdy wystąpi w tekście po raz drugi).

19: Pusty wiersz w tekście jest jednym ze sposobów informowania systemu o rozpoczęciu nowego akapitu. Pierwszy wiersz w nowym akapicie jest automatycznie wcinany. Wcięcie może być zniesione (wiersz 8) lub może być zmieniona jego szerokość (patrz str. viii).

31: Ustawienie numeru strony, tu na nr 7. Minus wskazuje, że numeracja ma być rzymska.

32: `\eject` łamie kolumnę, szanując wprowadzone instrukcje dotyczące pionowych odstępów.

33: `\end` kończy sesję T_EX-a, może być umieszczona w jakimkolwiek miejscu w pliku: T_EX nie czyta niczego poza nią. Jest to komenda o dużym priorytecie — przesuwa do góry zawartość kolumny. Jeśli zastosowano polecenie `\vfil`, należy użyć `\eject` (jak w wierszu 32), by `\end` nie zniósł ich działania.

Jeśli T_EX nie znajdzie komendy `\end` w pliku wejściowym, wyświetli znak `*`; należy ją wtedy wprowadzić z klawiatury.

JAK KORZYSTAĆ Z PODRĘCZNIKA

Przewodnik ten składa się z dwóch przeplatających się nawzajem części. Na stronach nieparzystych (po prawej), zwanych stronami wyjściowymi (*output pages*), przedstawiono podstawowe zastosowania programu; a dokładniej — zilustrowano te „rzeczy”, które można zrobić za pomocą T_EX-a. Na stronach parzystych (po lewej) znajdują się pudełka (*boxes*), które pokazują, co napisano, aby T_EX złożył stronę wyjściową (wynikową). Patrząc na pudełka wejściowe (*input boxes*), można od razu zobaczyć osiągnięty efekt składu, np. **pogrubiony** krój pisma, czy symbole \int lub \heartsuit , bez konieczności czytania długich wyjaśnień. Linijki tekstu w pliku wejściowym są dla wygody ponumerowane, a w miejscach wymagających wyjaśnienia dodano kilka uwag.

Przykłady zawarte w tej książce obejmują wiele standardowych zastosowań T_EX-a. Można korzystać z niej „wizualnie” — przeglądać kolejne strony do momentu, kiedy natrafi się na interesujące pojęcie, i wtedy szukać jego odpowiednika w pliku wejściowym (na stronie parzystej). Jeśli napotka się na problemy z odnalezieniem owego zagadnienia (ponieważ niemożliwe jest przedstawienie i omówienie wszystkiego, co *może* się wydarzyć), należy szukać go w dodatku. Znajduje się tam alfabetyczny spis wielu zastosowań programu i większości powszechnie wykorzystywanych komend.

```

1 \font\title=plssdc10 at 27,27pt % UWAGA: W TYM MIEJSCU ROZPOCZYNA SIĘ NOWY PLIK
2 \pageno=-9
3 \magnification=\magstephalf
4 \hsize 5,5 true in \hoffset ,125 true in
5 \vsize 8,5 true in \voffset ,1 true in
6 \parindent 35pt \parskip 1pt
7
8 \rightline{\title Przedmowa}%\title można wykorzystać po zdefiniowaniu (linia 1)
9 \vskip 2,5 true in
10 \noindent Główna część tego podręcznika została napisana w~lecie 1987 roku.
11 Byłem wówczas po doktoracie na Uniwerystecie Syracuse i~zaczynałem uczyć się
12 \TeX-a. W~celu lepszego zrozumienia programu zebrałem kilka przykładów jego
13 zastosowań. Przykłady te zainteresowały moich kolegów, tak więc dodałem kilka
14 uwag, dodatek, całość nazwałem {\sl An Absolute Beginner's Guide to Using
15 \TeX\} i~zrobiłem kilka kopii. Wydawało mi się, że to wystarczy.
16
17 Ale moi koledzy pokazali podręcznik swoim kolegom, którzy z~kolei przekazali
18 go dalej. Sposób nauki na przykładach bardzo się spodobał. W~ciągu następnych
19 czterech lat podręcznik nieformalnie krążył wśród użytkowników \TeX-a w~kilku
20 instytucjach; regularnie otrzymywałem prośby o~kolejne kopie. Częściowo właśnie
21 dzięki tym prośbom podręcznik, teraz znacznie poszerzony, został
22 opublikowany.\par Książka ta nie jest systematycznym przedstawieniem zastosowań
23 \TeX-a. Zawiera ilustracje tego, co można zrobić za pomocą \TeX-a, lecz
24 z~niewielką ilością objaśnień pracy programu; są to recepty na tworzenie
25 formatów różnego rodzaju, z~krótką analizą poszczególnych składników. Wyłącznie
26 zastosowania praktyczne, żadnej teorii. Takie podejście ma oczywiście wady, ale
27 ma też główną zaletę: pozwala nawet całkowitemu laikowi rozpocząć właściwe
28 użytkowanie \TeX-a, głównie przez kopiowanie przykładów.
29
30 Starałem się, aby przykłady obejmowały szeroki zakres zastosowań programu,
31 a~dodatek uwzględniał opisy wszystkich podstawowych komend. Użytkownicy,
32 których nie interesuje wewnętrzna praca \TeX-a, czy osiągnięcie niezwykłych, wręcz
33 ekstrawaganckich efektów składu, powinni znaleźć w~podręczniku wszystko, czego
34 potrzebują. Natomiast ci, którzy chcą wiedzieć więcej na temat samej pracy
35 programu, powinni szukać tej wiedzy w~innych źródłach. Najlepszym z~nich jest
36 książka Donalda Knutha {\sl The \TeX book}.\bigskip\bigskip
37 \line{\it Long Island, New York \hfil \sl Arvind Borde}
38 \line{\it August 1991 \hfil}
39 \vfil\eject

```

UWAGI

1: Komenda `\font` pozwala zdefiniować krój pisma do składu. `\title` umożliwia wybranie dużego pisma do tytułów sekcji. Komenda ta nie jest standardowa w \TeX -u, nie można jej wykorzystywać bez uprzedniego zdefiniowania. Więcej informacji dotyczących wykorzystania niestandardowych krojów można znaleźć w przykładzie X. Znak `%` oznacza komentarz: \TeX „nie widzi” tego, co znajduje się w wierszu za tym znakiem.

2: Komenda `\pageno` była objaśniona na str. vi. Domyślnie numery stron umieszczane są centrycznie, u dołu strony (na str. 2 pokazano, jak to można zmienić). Uwagi dotyczące str. xi zawierają więcej informacji o numerowaniu stron.

3: `\magnification` powiększa cały dokument; np. `\magstephalf` odpowiada powiększeniu o ok. 10%.

4: `\hsize` pozwala określić szerokość kolumny tekstu; słowo `true` oznacza rzeczywisty rozmiar, ignorując powiększenie; `in` odnosi się do cali. `\hoffset` przesuwa kolumnę tekstu względem standardowego marginesu strony — 1-calowego; parametr dodatni powoduje przesunięcie w prawo, ujemny w lewo.

5: `\vsize` i `\voffset` są pionowymi odpowiednikami komend omówionych powyżej. Parametr dodatni

`\voffset` przesuwa stronę w dół, ujemny — w górę. Górny margines ma domyślnie również 1 cal.

6: Słowo `pt` oznacza punkt. Jeden cal to 72,27 punktu. `\parindent` określa wcięcie akapitu (domyślnie 20pt). `\parskip` określa odległość między akapitami (domyślnie 0pt). Odstępy można „rozcigać”, jeśli to jest konieczne — patrz dodatek.

8: `\rightline` wyrównuje tekst do prawego marginesu.

9: `\vskip` składa odstęp pionowy o podanej wysokości.

14, 36, 37: `\sl` składa tekst pochyłą (*slanted*) odmianą pisma.

22: `\par` jest innym sposobem zakończenia akapitu.

36: `\bigskip` (patrz str. vi) nie wymaga umieszczenia w osobnym wierszu.

37, 38: `\line{ }` umieszcza zawartość nawiasów w jednym wierszu. `\hfil` jest poziomym odpowiednikiem komendy `\vfil`: wypełnia wiersz justunkiem. `\it` składa tekst kursywą.

39: `\eject` oznacza koniec kolumny tekstu na stronie. Zwykle poprzedzona komendą `\vfil`, wyrównuje zawartość do górnego marginesu, pozostawiając odstęp u dołu strony.

Przedmowa

Główna część tego podręcznika została napisana w lecie 1987 roku. Byłem wówczas po doktoracie na Uniwersytecie Syracuse i zaczynałem uczyć się $\text{T}_{\text{E}}\text{X}$ -a. W celu lepszego zrozumienia programu zebrałem kilka przykładów jego zastosowań. Przykłady te zainteresowały moich kolegów, tak więc dodałem kilka uwag, dodatek, całość nazwałem *An Absolute Beginner's Guide to Using $\text{T}_{\text{E}}\text{X}$* i zrobiłem kilka kopii. Wydawało mi się, że to wystarczy.

Ale moi koledzy pokazali podręcznik swoim kolegom, którzy z kolei przekazali go dalej. Sposób nauki na przykładach bardzo się spodobał. W ciągu następnych czterech lat podręcznik nieformalnie krążył wśród użytkowników $\text{T}_{\text{E}}\text{X}$ -a w kilku instytucjach; regularnie otrzymywałem prośby o kolejne kopie. Częściowo właśnie dzięki tym prośbom podręcznik, teraz znacznie poszerzony, został opublikowany.

Książka ta nie jest systematycznym przedstawieniem zastosowań $\text{T}_{\text{E}}\text{X}$ -a. Zawiera ilustracje tego, co można zrobić za pomocą $\text{T}_{\text{E}}\text{X}$ -a, lecz z niewielką ilością objaśnień pracy programu; są to recepty na tworzenie formatów różnego rodzaju, z krótką analizą poszczególnych składników. Wyłącznie zastosowania praktyczne, żadnej teorii. Takie podejście ma oczywiście wady, ale ma też główną zaletę: pozwala nawet całkowitemu laikowi rozpocząć właściwe użytkowanie $\text{T}_{\text{E}}\text{X}$ -a, głównie przez kopiowanie przykładów.

Starałem się, aby przykłady obejmowały szeroki zakres zastosowań programu, a dodatek uwzględniał opisy wszystkich podstawowych komend. Użytkownicy, których nie interesuje wewnętrzna praca $\text{T}_{\text{E}}\text{X}$ -a, czy osiąganie niezwykłych, wręcz ekstrawaganckich efektów składu, powinni znaleźć w podręczniku wszystko, czego potrzebują. Natomiast ci, którzy chcą wiedzieć więcej na temat samej pracy programu, powinni szukać tej wiedzy w innych źródłach. Najlepszym z nich jest książka Donalda Knutha *The $\text{T}_{\text{E}}\text{X}$ book*.

*Long Island, New York
August 1991*

Arvind Borde

```

1 \pageno=-11
2 \rightline{\title Spis treści}%UWAGA: \title nie jest standardową komendą.
3 \vskip 2,3 true in
4 \noindent Jak korzystać z podręcznika\dotfill vii\break
5 {\sl Zadany format Plain \TeX-a; zwykle odstępy; wyśrodkowanie.}\bigskip
6 \noindent Przedmowa\dotfill ix\break
7 {\sl Powiększanie; adiustacja rozmiaru i~położenia strony; akapity; łamanie
8 stron.}\bigskip
9 \noindent Spis treści\dotfill xi\break
10 {\sl Tworzenie list wyliczanych; podział wierszy.}\bigskip
11 \noindent Wstęp\dotfill 1\break
12 {\sl Podstawowe komendy; komendy znakowe; kroje pisma; odstępy;}\hfil\break
13 {\sl tryb matematyczny; przetwarzanie pliku wejściowego, błędy.}\bigskip
14 \noindent Przykłady\medskip
15 \item{I.} Prosto o piśmiennictwie\dotfill 21\break
16 {\sl Eksponowanie tekstu w~wąskim lub szerokim formacie;}\hfil\break
17 {\sl modelowanie akapitów.}\medskip
18 \item{II.}  $\sqrt{2}$  --- liczba niewymierna\dotfill 25\break
19 {\sl Eksponowanie równań; ułamki; potęgi; pierwiastki; przypisy.}\medskip
20 \item{III.} Poziomy Cauchy'ego\dotfill 31\break
21 {\sl Symbole teorii zbiorów; puste odstępy na schematy.}\medskip
22 \item{IV.} Hiszpańskie interludium\dotfill 35\break
23 {\sl Niezwykła interpunkcja; akcenty; komendy nakładania.}\medskip
24 \item{V.} Spułapkowane powierzchnie w~przestrzeni  $G\{\text{"o}\}$ dla\dotfill 37\break
25 {\sl Macierze; justowanie równań; indeksy; pochodne cząstkowe; przypisy.}
26 \medskip
27 \item{VI.} Sunil Gavaskar: a~Statistical Tribute\dotfill 45\break
28 {\sl Tabele.}\medskip
29 \item{VII.} Zerowe rozwiązania równania  $\ddot{x} + F(t)x = 0$ \dotfill
30 53\break{\sl Pochodne i~całki; justowanie równań.}\medskip

```

UWAGI

Komendy `\pageno`, `\rightline`, `\title`, `\%`, `\vskip`, `\noindent` oraz `\sl` były omówione na poprzednich stronach.

1: Numerowanie stron w \TeX -u jest automatyczne i zwykle nie musimy ponownie ustawiać numeru każdej nowej strony. W tym przypadku, ponieważ strony „prawe” i „lewe” pochodzą z osobnych plików, numeracja każdej nowej strony nieparzystej musi wzrastać o 2. Można tak skonstruować komendę, że program będzie to robił automatycznie, ale tutaj ręczne numerowanie stron początkowych jest prostsze. Na str. 2 będzie pokazany niestandardowy, automatyczny sposób numeracji stron.

4, 6, itd.: `\dotfill` wypełnia pustą przestrzeń wiersza kropkami. `\break` informuje o końcu wiersza. Wiersz następujący zaraz za komendą jest traktowany jako część tego samego akapitu i nie jest wcinany, więc `\noindent` jest w tym przypadku zbędne.

12: `\hfil\break` dokonuje podziału wiersza. `\hfil` wypełnia pozostałą część linii justunkiem, a `\break` faktycznie ją łamie.

14, 17: `\medskip` składa średni odstęp pionowy. Nie ma znaczenia, czy umieszczamy tę komendę w osobnym wierszu, czy nie.

15, itd.: `\item` służy do składu wyliczeń. Główny tekst listy jest wcięty na odległość równą bieżącej wartości wcięcia akapitu (patrz `\parindent` na poprzedniej stronie), liczby zaś lub inne znaczniki identyfi-

kujące jednostki listy są umieszczane na marginesie tekstu, z wyrównaniem do prawej strony wcięcia. Materiał jest traktowany jako całość, dopóki \TeX nie natopka kolejnej komendy `\item` lub komendy kończącej akapit.

18: `\sqrt{2}` składa symbol pierwiastka kwadratowego. Komenda ta działa wyłącznie w trybie matematycznym, tzn. musi być otoczona znakami `$`.

24: Akcentowany znak `ö` jest składany komendą `\"o`.

29: `\ddot{x}` składa znak \ddot{x} w trybie matematycznym. Jest wiele komend, które działają wyłącznie w tym trybie, dlatego należy zwracać uwagę, jaką komendę i w jakim trybie wykorzystujemy. (W trybie matematycznym obowiązują inne reguły dotyczące odstępów niż w trybie tekstowym.)

30: Komenda `\break` może wystąpić w środku wiersza w pliku wejściowym i powoduje podział wiersza tak samo, jak w przypadku gdy jest umieszczona na jego końcu. \TeX sam podejmuje decyzję, w którym miejscu zakończyć stronę; nie jest wymagana w tym celu specjalna komenda.

KOMENTARZ

Jak można było zauważyć do tej pory, \TeX oferuje użytkownikowi dużą dowolność w aranżowaniu pliku wejściowego, bez oddziaływania na skład. Zaleta ta powoduje, że edycja i zmiany w plikach stają się niezwykle łatwe.

Spis treści

Jak korzystać z podręcznika	vii
<i>Zadany format Plain T_EX-a; zwykle odstępy; wyśrodkowanie.</i>	
Przedmowa	ix
<i>Powiększanie; adiustacja rozmiaru i położenia strony; akapity; łamanie stron.</i>	
Spis treści	xi
<i>Tworzenie list wyliczanych; podział wierszy.</i>	
Wstęp	1
<i>Podstawowe komendy; komendy znakowe; kroje pisma; odstępy; tryb matematyczny; przetwarzanie pliku wejściowego, błędy.</i>	
Przykłady	
I. Prosto o piśmiennictwie	21
<i>Eksponowanie tekstu w wąskim lub szerokim formacie; modelowanie akapitów.</i>	
II. $\sqrt{2}$ — liczba niewymierna	25
<i>Eksponowanie równań; ułamki; potęgi; pierwiastki; przypisy.</i>	
III. Poziomy Cauchy’ego	31
<i>Symbole teorii zbiorów; puste odstępy na schematy.</i>	
IV. Hiszpańskie interludium	35
<i>Niezwykła interpunkcja; akcenty; komendy nakładania.</i>	
V. Spuławpkowane powierzchnie w przestrzeni Gödla	37
<i>Macierze; justowanie równań; indeksy; pochodne cząstkowe; przypisy.</i>	
VI. Sunil Gavaskar: a Statistical Tribute	45
<i>Tabele.</i>	
VII. Zerowe rozwiązania równania $\ddot{x} + F(t)x = 0$	53
<i>Pochodne i całki; justowanie równań.</i>	

1	<code>\item{VIII.} Wiele imion Diraka\dotfill</code>	59\break
2	<code>{\sl Wcięcia zawieszane.}</code>	
3	<code>\medskip</code>	
4	<code>\item{IX.} Zlistów Ramanujana do Hardy’ego\dotfill</code>	61\break
5	<code>{\sl Szczegóły składu równań.}</code>	
6	<code>\medskip</code>	
7	<code>\item{X.} Typografia\dotfill</code>	65\break
8	<code>{\sl Odmiany i~stopnie krojów pisma.}</code>	
9	<code>\medskip</code>	
10	<code>\item{XI.} List\dotfill</code>	67\break
11	<code>{\sl Definicje; listy.}</code>	
12	<code>\medskip</code>	
13	<code>\item{XII.} Notka\dotfill</code>	69\break
14	<code>{\sl Więcej definicji; przechowywanie zmiennych; komendy pętli; notki.}</code>	
15	<code>\medskip</code>	
16	<code>\item{XIII.} Konwersacja\dotfill</code>	71\break
17	<code>{\sl Komendy warunkowe; prosty scenariusz.}</code>	
18	<code>\medskip</code>	
19	<code>\item{XIV.} Ostatnie twierdzenia Fermata\dotfill</code>	73\break
20	<code>{\sl Programowanie w~\TeX-u.}</code>	
21	<code>\medskip</code>	
22	<code>\item{XV.} O~boksie\dotfill</code>	75\break
23	<code>{\sl Umieszczanie materiału w~pionowych i~poziomych pudełkach.}</code>	
24	<code>\medskip</code>	
25	<code>\item{XVI.} Ruch dookoła\dotfill</code>	77\break
26	<code>{\sl Obramowanie tekstu; cieniowanie pudełek; inne ciekawe efekty.}</code>	
27	<code>\medskip</code>	
28	<code>\item{XVII.} Sceny z czasopisma\dotfill</code>	81\break
29	<code>{\sl Proste aspekty składu czasopism.}</code>	
30	<code>\bigskip</code>	
31	<code>\noindent Podziękowania\dotfill</code>	87\break
32	<code>{\sl Przełączanie fontów; skład „,w~chorągiewkę’’; zakaz podziału</code>	
33	<code>wyrazów.} \bigskip</code>	
34	<code>\noindent Cytowana literatura\dotfill</code>	89\break
35	<code>{\sl Automatyczne numerowanie; listy wewnątrz list.}</code>	
36	<code>\bigskip</code>	
37	<code>\noindent Dodatek\dotfill</code>	91\break
38	<code>{\sl Skorowidz i~słownik.}</code>	
39	<code>\bigskip</code>	
40	<code>\noindent Uwagi końcowe\dotfill</code>	161\break
41	<code>{\sl Plik wejściowy do pliku wejściowego (czyli jak złożyć tę książkę).}</code>	
42	<code>\pageno=-13</code>	
43	<code>\bye</code>	

UWAGI

KOMENTARZ

Jak stwierdzono w uwagach do poprzedniej strony, użytkownik ma możliwość dowolnego aranżowania pliku wejściowego. Wynika to m.in. z faktu, iż wielokrotne spacje w pliku wejściowym są traktowane przez system jako jedna (oczywiście jest różnica pomiędzy brakiem spacji a jedną spacją). Można sądzić, że utrzymywanie pliku wejściowego w należyтым porządku (jak u góry strony) nie jest warte wysiłku. Niemniej jednak np. umieszczanie komend odstepu pionowego w osobnych wierszach pomaga w czytaniu pliku wejściowego.

1: Wyliczenie jest kontynuowane z poprzedniej strony bez żadnych dodatkowych działań ze strony użytkownika.

31, 34, 37, 40: Komenda `\noindent` jest potrzebna po komendzie `\bigskip`, ponieważ \TeX traktuje taki odstęp pionowy jako rozpoczęcie akapitu, a w tym miejscu wcięte wiersze są niepożądane.

42: Strona po raz kolejny jest numerowana ręcznie. Zwykle nie jest to konieczne — patrz objaśnienie do pierwszego wiersza na str. x.

43: `\bye` jest jednym ze sposobów wskazywania końca pliku. Jest to skrót kombinacji komend skonstruowanych z udziałem komendy `\end`, tak więc wykonuje coś więcej niż tylko nagłe zakończenie sesji programu. Szerzej zostanie to omówione w dodatku. Ten podręcznik był składany z kilku osobnych plików, tak więc użytkownik spotka się jeszcze kilka razy (przed właściwym końcem) z komendą `\bye`.

VIII.	Wiele imion Diraka	59
	<i>Wcięcia zawieszane.</i>	
IX.	Z listów Ramanujana do Hardy’ego	61
	<i>Szczegóły składu równań.</i>	
X.	Typografia	65
	<i>Odmiany i stopnie krojów pisma.</i>	
XI.	List	67
	<i>Definicje; listy.</i>	
XII.	Notka	69
	<i>Więcej definicji; przechowywanie zmiennych; komendy pętli; notki.</i>	
XIII.	Konwersacja	71
	<i>Komendy warunkowe; prosty scenariusz.</i>	
XIV.	Ostatnie twierdzenia Fermata	73
	<i>Programowanie w T_EX-u.</i>	
XV.	O boksie	75
	<i>Umieszczanie materiału w pionowych i poziomych pudełkach.</i>	
XVI.	Ruch dookoła	77
	<i>Obramowanie tekstu; cieniowanie pudełek; inne ciekawe efekty.</i>	
XVII.	Sceny z czasopisma	81
	<i>Proste aspekty składu czasopism.</i>	
	Podziękowania	87
	<i>Przełączanie fontów; skład „w chorągiewkę”; zakaz podziału wyrazów.</i>	
	Cytowana literatura	89
	<i>Automatyczne numerowanie; listy wewnątrz list.</i>	
	Dodatek	91
	<i>Skorowidz i słownik.</i>	
	Uwagi końcowe	161
	<i>Plik wejściowy do pliku wejściowego (czyli jak złożyć tę książkę).</i>	

```

1 \font\title=plssdc10 scaled 2488 % Początek nowego pliku.
2 \magnification=\magstep1
3 \baselineskip 22 true pt \parskip=0pt plus3pt
4 \hsize 5,5 true in \hoffset ,125 true in
5 \vsize 9,45 true in \voffset ,1 true in
6
7 \rightline{\title Wstęp} % UWAGA: ,,title'' nie jest standardową komendą.
8 \rightline{\sl Podstawowe komendy; komendy znakowe; kroje pisma; odstępy;}
9 \line{\hfil\sl tryb matematyczny; przetwarzanie pliku wejściowego; błędy.}
10 \vskip 2,5 true in
11 \vskip-2\baselineskip
12 \noindent \TeX\ --- powszechnie wymawiany ,,tech'' --- jest programem napisanym
13 przez Donalda Knutha w celu ,,składu pięknych książek''. Jest to mądry
14 i~potężny program, za jego pomocą można zrobić bardzo wiele rzeczy. Ma on
15 jednak reputację programu trudnego. Częściowo dlatego, {\sl ponieważ\} jest
16 tak potężny, ale częściowo dlatego, że entuzjaści \TeX-a czasem próbują
17 opowiedzieć osobie rozpoczynającej pracę z~\TeX-em od razu wszystko na temat
18 programu: {\it wszystko}, co może zrobić, i~jak pracuje. To powoduje, że
19 program odstrasza potencjalnych użytkowników --- a~szkoda, ponieważ naprawdę
20 nie jest trudno złożyć za pomocą \TeX-a prostego, tzw. gładkiego tekstu.
21
22 Książkę tę napisano, aby pokazać, jak łatwo można tworzyć proste dokumenty za
23 pomocą programu \TeX. Na str. {\sl vii\} znajduje się krótki opis pracy
24 z~podręcznikiem. \underbar{Przeczytaj go.} Chociaż dalsze przykłady w~książce
25 wydają się nieco bardziej skomplikowane, w~większości przypadków dotyczą
26 prostych zastosowań programu. Wszystkie wyrafinowane szczegóły, które
27 opuszczono, można znaleźć w~książce dotyczącej \TeX-a, napisanej przez
28 D.~Knutha$^1$.
29
30 \TeX\ jest programem {\sl do składu\}; można wykorzystać go do umieszczenia
31 tekstu i~innych elementów w~określonych pozycjach na drukowanej stronie. Aby
32 to zrobić, najpierw należy stworzyć plik komputerowy zawierający materiał, który
33 chcemy wydrukować. Można wykorzystać do tego {\it dowolny\} edytor tekstu.
34 (Zakładam, że użytkownik wie, jak stworzyć i~wyedytować plik na swoim
35 komputerze. Jeśli nie, powinien zapytać kogoś, jak to należy zrobić.)

```

UWAGI

1: `\font` pozwala zdefiniować nazwy krojów pisma. Patrz str. vii. `\title` w tym miejscu określa krój pisma dużego stopnia do tytułów sekcji (wyrażenie po prawej stronie znaku = może być dowolnie zmienione). Patrz przykład X.

2–5: W tych wierszach zdefiniowany jest format stron nieparzystych.

2: `\magnification` — tutaj `\magstep1`, komenda powiększająca cały dokument (stopień pisma, odstępy, itd.) 1,2 raza. Możliwe są również dalsze powiększenia — `\magstep2`, ..., `\magstep5`. Każdy krok powiększenia jest 1,2 raza większy od poprzedniego.

Nie można stosować różnych powiększeń tego typu wewnątrz tego samego dokumentu.

3: `\baselineskip` określa odstęp między wierszami. Wartość domyślna tego parametru wynosi 12pt, gdzie pt oznacza punkt. Jeden cal ma 72,27 punktu. true oznacza tutaj „ignoruj powiększenie”; bez tej komendy wyspecyfikowanie rozmiaru 22 punktów dałoby w rzeczywistości 26,4 (= 22 × 1,2) punktu na stronach składu (stronach wynikowych). `\parskip` określa odstęp pomiędzy akapitami; plus umożliwia elastyczne zwiększenie tej odległości, jeśli to jest konieczne. Zdolność „rozciągania” odstępów jest bardzo istotna przy wyrównywaniu do dolnych i prawych marginesów.

4, 5: Określenie rozmiaru strony, tak jak na str. viii; in oznacza cale.

7–9: `\rightline{X}` i `\line{\hfil X}` to dwa różne sposoby na dosunięcie parametru X do prawego marginesu. `\sl` składa tekst odmianą pochylą.

10, 11: Pierwsza komenda `\vskip` składa 2,5-calowy pionowy odstęp pod ostatnio drukowanym wierszem. Aby tekst na każdej stronie rozpoczynał się na tej samej wysokości co na stronie tytułowej, zastosowano kolejny `\vskip`, kompensujący dwie linie podtytułu na tej stronie.

12, 19: --- składa długą pauzę — (myślnik).

13: Zauważ, jak się składa cudzysłowy (prawy i lewy).

15, 23, 30: `\sl` składa tekst pochylą odmianą. Komenda `\` jest poprawką do pisma pochylego (tzw. korekta kursywy, czyli mały odstęp zapobiegający zachodzeniu kursywy lub pisma pochylego na odstęp przed następnym znakiem).

18, 33: `\it` włącza kursywę.

24: `\underbar` tworzy podkreślenie tekstu (styl rzadko stosowany w profesjonalnym składzie). Nawiasy grupujące { i } określają, jaka część tekstu ma zostać podkreślona. Nawiasy te wykorzystywane są trochę inaczej do „włączania” kroju pisma, jak np. `{\sl vii\}` w linii powyżej. Szerzej zostało to omówione w dodatku, pod hasłem argumenty.

28: `^` jest komendą składającą potęgę w trybie matematycznym; służy też do składu odnośników. Znak \$ włącza/wyłącza tryb matematyczny.

Wstęp

*Podstawowe komendy; komendy znakowe; kroje pisma; odstępy;
tryb matematyczny; przetwarzanie pliku wejściowego; błędy.*

T_EX — powszechnie wymawiany „tech” — jest programem napisanym przez Donalda Knutha w celu „składu pięknych książek”. Jest to mądry i potężny program, za jego pomocą można zrobić bardzo wiele rzeczy. Ma on jednak reputację programu trudnego. Częściowo dlatego, *ponieważ* jest tak potężny, ale częściowo dlatego, że entuzjaści T_EX-a czasem próbują opowiedzieć osobie rozpoczynającej pracę z T_EX-em od razu wszystko na temat programu: *wszystko*, co może zrobić, i jak pracuje. To powoduje, że program odstrasza potencjalnych użytkowników — a szkoda, ponieważ naprawdę nie jest trudno złożyć za pomocą T_EX-a prostego, tzw. gładkiego tekstu.

Książkę tę napisano, aby pokazać, jak łatwo można tworzyć proste dokumenty za pomocą programu T_EX. Na str. vii znajduje się krótki opis pracy z podręcznikiem. [Przeczytaj go](#). Chociaż dalsze przykłady w książce wydają się nieco bardziej skomplikowane, w większości przypadków dotyczą prostych zastosowań programu. Wszystkie wyrafinowane szczegóły, które opuszczono, można znaleźć w książce dotyczącej T_EX-a, napisanej przez D. Knutha¹.

T_EX jest programem *do składu*; można wykorzystać go do umieszczenia tekstu i innych elementów w określonych pozycjach na drukowanej stronie. Aby to zrobić, najpierw należy stworzyć plik komputerowy zawierający materiał, który chcemy wydrukować. Można wykorzystać do tego *dowolny* edytor tekstu. (Zakładam, że użytkownik wie, jak stworzyć i wyedytować plik na swoim komputerze. Jeśli nie, powinien zapytać kogoś, jak to należy zrobić.)


```

1 Trzeba uważać jednak, by nie używać żadnych {\sl formatujących\/} komend
2 stosowanych w danym edytorze. Na przykład, jeśli chcemy zakończyć stronę
3 w~określonym punkcie, musimy wykorzystać w tym celu komendę \TeX-a, nie zaś
4 komendę specyficzną dla edytora. Plik wejściowy można utworzyć na dowolnym
5 komputerze, nawet jeśli nie jest na nim zainstalowany \TeX. Komendy \TeX-a
6 potrzebne do stworzenia interesującego nas formatu wprowadzamy z~klawiatury.
7 Na stronach parzystych tej książki pokazano, jak wyglądają takie pliki wejściowe
8 ({\it input files\/}). Kiedy mamy gotowy plik, możemy go przetworzyć za pomocą
9 \TeX-a. (Jeśli plik powstał w~systemie, w~którym brak \TeX-a, należy przenieść
10 go na inny komputer, z~zainstalowanym programem \TeX.) Program tworzy osobny
11 plik, który można przesłać do drukarki. Proces przetwarzania tekstu jest
12 omawiany powtórnie na str.~17.
13
14 Jedną z najpotężniejszych zalet \TeX-a jest fakt, iż pozwala on użytkownikowi
15 na swobodę w~definiowaniu komend. Na przykład, jeśli istnieje kombinacja kilku
16 komend, którą często wykorzystujemy, można zdefiniować jedną komendę, która
17 będzie pracować jak cały zestaw. Istnieje wiele pakietów komend i~definicji
18 (czasem nazywanych makrami) wykorzystywanych do specjalnych celów.
19 W rzeczywistości pod pojęciem „\TeX” kryje się kombinacja {\sl podstawowych\/}
20 komend --- są to te wyrażenia pierwotne, które tworzą \TeX-a --- i~zestaw komend
21 dodatkowych, zdefiniowanych za pomocą pojęć pierwotnych. Właściwa nazwa całego
22 pakietu to Plain \TeX, ale generalnie będziemy mówić (tak jak większość
23 użytkowników) po prostu \TeX.
24
25 %OTO ZESTAW ZAAWANSOWANYCH KOMEND, POTRZEBNYCH DO UMIESZCZENIA NUMERU STRONY
26 %W PRAWYM GÓRNYM ROGU STRONY SKŁADU. JEŚLI DOPIERO ROZPOCZYNASZ PRACĘ Z \TeX-em
27 %MOŻESZ JE ZIGNOROWAĆ.
28 \nopagenumbers
29 \def\newpageno {\multiply\pageno by2 \advance\pageno by-1
30 \rlap{\hbox to 1,125 true in{\hfil\bf\the\pageno}}}}
31 \headline{\hfil\it Wstęp\newpageno}
32 %KONIEC BZDUR. MOŻESZ ZNOWU „OTWORZYĆ OCZY”.

```

UWAGI

Niektóre z komend pojawiły się już wcześniej: `\sl`, `\bf` i `\it`. Komenda `\/` daje poprawkę do pisma pochylego (tzw. korektę kursywy/italikową), dodatkowy odstęp w miejscu, w którym kończy się skład kursywą lub odmianą pochylą. Komenda `\TeX` składa logo „TEX”. Uwagi do str. 5 zawierają więcej informacji na temat tych komend.

6, 12, itd.: Komenda `~` składa wiązanie (*tie*) — pojedynczą spację, w miejscu której wiersz nie może zostać podzielony. Wykorzystanie jej w linii 12 daje pewność, że „17” nie pojawi się na początku wiersza. Dobrym sposobem jest automatyczne wprowadzanie takich łączy, np. po spójnikach (i, a). W dodatku pod hasłem **podział wierszy** znajduje się więcej informacji na ten temat.

PRZEPROSINY

Komendy w liniach 28–31 dotyczą definicji numeracji stron nieparzystych w pozostałej części książki. Numery stron będą się pojawiały teraz w prawym górnym rogu, będą złożone krojem pogrubionym i będą miały wyłączanie nieparzyste wartości. Strony parzyste są numerowane za pomocą podobnej komendy, przedstawionej w uwagach końcowych. Osobne komendy dotyczą numeracji stron parzystych i nieparzystych, ponieważ książka ta powstała z kilku oddzielnie prze-

twarzanych plików. Komendy formatujące do dodatku (odtworzone w uwagach końcowych) pokazują, jak łączyć te dwie komendy w jedną.

Linie 25–27 i 32 są komentarzem (są poprzedzone znakiem `%`). *Wszystkie komendy zawarte w liniach 28–31 omówiono w dalszej części podręcznika, zwłaszcza w dodatku.* W tym miejscu jedynie krótkie wyjaśnienie:

28: `\nopagenumbers` „wylacza” standardowy styl numerowania stron, w którym sukcesywnie wzrastające numery pojawiają się centrycznie u dołu strony (patrz str. 1).

29: Definicja komendy `\newpageno`; normalny numer strony — określony standardową komendą `TEX-a, \pageno` — jest mnożony przez 2, a następnie pomniejszony o 1. To daje właściwy numer strony.

30: W wierszu tym znajduje się dalszy ciąg definicji komendy `\newpageno`; określenie pozycji, odstępów i kroju pisma oraz wybranie nowej wartości numeru strony z `\pageno`.

31: Komenda `\headline` dotyczy nagłówka strony. Tutaj składa się on ze słowa „Wstęp”, złożonego kursywą, dosuniętego do prawego marginesu, z nowym numerem strony po prawej. Na następnej stronie umieszczone są uwagi dotyczące nagłówków.

Trzeba uważać jednak, by nie używać żadnych *formatujących* komend stosowanych w danym edytorze. Na przykład, jeśli chcemy zakończyć stronę w określonym punkcie, musimy wykorzystać w tym celu komendę `TEX-a`, nie zaś komendę specyficzną dla edytora. Plik wejściowy można utworzyć na dowolnym komputerze, nawet jeśli nie jest na nim zainstalowany `TEX`. Komendy `TEX-a` potrzebne do stworzenia interesującego nas formatu wprowadzamy z klawiatury. Na stronach parzystych tej książki pokazano, jak wyglądają takie pliki wejściowe (*input files*). Kiedy mamy gotowy plik, możemy go przetworzyć za pomocą `TEX-a`. (Jeśli plik powstał w systemie, w którym brak `TEX-a`, należy przenieść go na inny komputer, z zainstalowanym programem `TEX`.) Program tworzy osobny plik, który można przesłać do drukarki. Proces przetwarzania tekstu jest omawiany powtórnie na str. 17.

Jedną z najpotężniejszych zalet `TEX-a` jest fakt, iż pozwala on użytkownikowi na swobodę w definiowaniu komend. Na przykład, jeśli istnieje kombinacja kilku komend, którą często wykorzystujemy, można zdefiniować jedną komendę, która będzie pracować jak cały zestaw. Istnieje wiele pakietów komend i definicji (czasem nazywanych makrami) wykorzystywanych do specjalnych celów. W rzeczywistości pod pojęciem „`TEX`” kryje się kombinacja *podstawowych* komend — są to te wyrażenia pierwotne, które tworzą `TEX-a` — i zestaw komend dodatkowych, zdefiniowanych za pomocą pojęć pierwotnych. Właściwa nazwa całego pakietu to Plain `TEX`, ale generalnie będziemy mówić (tak jak większość użytkowników) po prostu `TEX`.

W podręczniku tym pokazano, jak wykorzystywać komendy formatu Plain `TEX`, podstawowy zestaw makr `TEX-a`. Komendy te nie zawsze będą działać w ten sam sposób; mogą nie być rozpoznawane przez system, jeśli pracuje się z innymi pakietami. Ilustracje sposobów tworzenia nowych komend znajdują się w przykładach (począwszy od X). Do tego miejsca nie wprowadzono żadnych komend specjalnych, z dwoma wyjątkami (dwie specjalne komendy już się pojawiły). Wyjątkami tymi są: zdefiniowany krój pisma dużego stopnia do tytułów (na początku pliku wejściowego str. 1) i komenda definiująca niestandardową numerację stron nieparzystych (w pliku wejściowym

```

1 W~podręczniku tym pokazano, jak wykorzystywać komendy formatu Plain \TeX,
2 podstawowy zestaw makr \TeX-a. Komendy te nie zawsze będą działać w~ten sam
3 sposób; mogą nie być rozpoznawane przez system, jeśli pracuje się z~innymi
4 pakietami. Ilustracje sposobów tworzenia nowych komend znajdują się
5 w~przykładach (począwszy od~X). Do tego miejsca nie wprowadzono żadnych komend
6 specjalnych, z~dwoma wyjątkami (dwie specjalne komendy już się pojawiły).
7 Wyjątkami tymi są: zdefiniowany krój pisma dużego stopnia do tytułów (na
8 początku pliku wejściowego str.~1) i~komenda definiująca niestandardową
9 numerację stron nieparzystych (w~pliku wejściowym umieszczona przed bieżącym
10 akapitem). Komendy te zostały zdefiniowane (i~wykorzystane) na początku, ale
11 definicje te należy wyjaśnić trochę dalej. Na niektóre z komend
12 przedstawionych w~przykładach X--XVII należy zwrócić szczególną uwagę: komendy
13 formatujące listy i~notki, komendy tworzące ramki wokół tekstu, itd. Główne
14 zasady działania tych komend są zaprezentowane także w dodatku.
15
16 Ponieważ książka ta ma uczyć na przykładach, w większości składa się z~krótkich
17 rozdziałów ilustrujących możliwości \TeX-a. Przed przystąpieniem do tych
18 rozdziałów, pożyteczne będzie przeczytanie pozostałej części wstępu.
19 Przedstawiono tam, bardzo zwięźle, podstawowe cechy programu.
20 \bigskip
21 \centerline{${\star}\quad{\star}\quad{\star}$}
22 \vfil\eject

```

UWAGI

1: Jeśli po logo \TeX nie następuje od razu jakiś znak, nie będący literą, np. znak interpunkcji, należy wstawić po komendzie $\backslash\TeX$ kolejny *backslash* \backslash i spację, aby zapobiec „połknięciu” znajdującego się za nazwą komendy odstępu. Może to zabrzmieć dziwnie, ale przyczyna jest prosta: w momencie gdy program „czyta” komendę, chce wiedzieć, kiedy jej nazwa się kończy; według zasad programu jednym z takich wskaźników jest spacja, stąd może ona być potraktowana w tym przypadku nie jako instrukcja *pozostaw spację*, ale jako wskaźnik końca nazwy. Innym sposobem wyraźnego wskazania końca nazwy komendy jest wykorzystanie nawiasów, np. $\{\backslash\TeX\}$ lub $\backslash\TeX\{\}$.

21: $\backslash\text{centerline}$ oznacza wyśrodkowanie tekstu. $\backslash\text{star}$ jest komendą z trybu matematycznego, składa znak \star . Znak $\$$ włącza i wyłącza tryb matematyczny. Komenda $\backslash\text{quad}$ składa mały odstęp; działa zarówno w trybie matematycznym, jak i tekstowym.

GŁÓWNE UWAGI DOTYCZĄCE NAGŁÓWKÓW

Z uwag do poprzedniej strony wiemy już, że komenda $\backslash\text{headline}$ składa nagłówek strony: wiersz tekstu,

który będzie się powtarzać u góry każdej strony, do momentu zmiany definicji tej komendy.

Przy wprowadzaniu nagłówków należy być ostrożnym z dwóch przyczyn. Po pierwsze, w przypadku braku wyraźnego określenia kroju pisma może zdarzyć się, że \TeX zastosuje w nagłówkach krój aktualnie używany.

Drugą ważną sprawą jest fakt, iż \TeX „czyta” tekst gromadząc w pamięci więcej niż jedną stronę materiału, zanim wybierze właściwe miejsce do złamania strony. Kiedy tworzy układ strony, wykorzystuje ostatnio czytaną instrukcję dotyczącą składu nagłówka. Niekiedy może się zdarzyć, że przeczyta nową instrukcję dotyczącą materiału, który znajdzie się dopiero na następnej stronie, a na bieżącej umieści niewłaściwy nagłówek. Aby zapobiec takim problemom, \TeX znaczy tekst i „odkłada go na bok” do użycia *po* umieszczeniu strony. Przy takim podejściu nagłówki są składane dużo poprawniej. Zobacz objaśnienie komendy $\backslash\text{mark}$ w dodatku.

Dyskusja ta odnosi się także do przypisów.

umieszczona przed bieżącym akapitem). Komendy te zostały zdefiniowane (i wykorzystane) na początku, ale definicje te należy wyjaśnić trochę dalej. Na niektóre z komend przedstawionych w przykładach X–XVII należy zwrócić szczególną uwagę: komendy formatujące listy i notki, komendy tworzące ramki wokół tekstu, itd. Główne zasady działania tych komend są zaprezentowane także w dodatku.

Ponieważ książka ta ma uczyć na przykładach, w większości składa się z krótkich rozdziałów ilustrujących możliwości \TeX -a. Przed przystąpieniem do tych rozdziałów, pożyteczne będzie przeczytanie pozostałej części wstępu. Przedstawiono tam, bardzo zwięźle, podstawowe cechy programu.

★ ★ ★

```

1 \noindent{\bf \S\ Komendy znakowe}\hfil\break
2 Plik przetwarzany w \TeX-u jest napisany przy użyciu standardowych znaków
3 z~klawiatury komputera. Większość tych znaków będzie drukowana w~pliku
4 wynikowym w~takiej postaci, w~jakiej występuje na ekranie. Następujące
5 symbole mają jednak specjalne znaczenie jako komendy \TeX-a:
6
7 \centerline {\tt \string\ \qqquad \string{ \qqquad \string} \qqquad \$ \qqquad &
8 \qqquad ^ \qqquad \string_ \qqquad \% \qqquad ~ \qqquad \#}
9
10 \noindent Aby {\it je\/} otrzymać w~takiej postaci na wydruku, należy zastosować
11 specjalne instrukcje. Linie 7--8 w~pliku wejściowym i~uwagi poniżej pudełka
12 z~plikiem wejściowym pokazują, jak to zrobić. Podstawowe funkcje wymienionych
13 powyżej symboli:
14 \smallskip
15 {\narrower \parindent 25 true pt
16 \item{\tt\string\ } oznacza początek wszystkich komend (z~wyjątkiem niektórych
17 wymienionych powyżej);
18 \item{\tt\string{\quad\string}} grupują wyrażenia, informują program,
19 którą komendę zastosować do którego fragmentu tekstu;
20 \item{\tt\$} oznacza początek i~koniec trybu matematycznego;
21 \item{\tt&} składa tekst, liczby itp. w~kolumnach (np. w~tabeli lub macierzy),
22 oznacza skok do następnej kolumny;
23 \item{\tt^} składa frakcję górną w~trybie matematycznym;
24 \item{\tt\_} składa frakcję dolną w~trybie matematycznym;
25 \item{\tt%} określa komentarz w~pliku wejściowym, czyli to, co nie powinno
26 znaleźć się w~składzie; jeśli program napotka taki znak, kończy czytanie danego
27 wiersza i przeskakuje do następnego;
28 \item{\tt~} oznacza pojedynczą, niełamliwą spację (w~tym miejscu nie
29 nastąpi podział wiersza);
30 \item{\tt\#} oznacza numer parametru w~komendzie ogólnej, program może później
31 wstawić w~to miejsce np. tekst.
32 \smallskip}
33 \noindent Symbole te pojawią się jeszcze w~pozostałej części podręcznika.
34 Istnieje możliwość ich przededefiniowania, tak aby inne znaki pełniły funkcję
35 komend; zostanie to omówione w~dodatku.
36 \vfil\ejct

```

UWAGI

1: Komendy `\noindent`, `\bf` i `\hfil\break` były już wcześniej omawiane. Komenda `\S` składa symbol §. (Drugi znak `\` daje pewność, że po symbolu § znajduje się odstęp.)

7–8: `\centerline` wyśrodkowuje tekst. `\tt` składa tekst maszynowym krojem pisma, w niniejszym podręczniku wykorzystywanym do składu komend. `\string` jest specyficzną komendą, określa *verbatim* (dosłowny skład) bezpośrednio następującego znaku. Umożliwia skład nazw komend \TeX -owych w tekście (wykorzystana w niniejszym podręczniku). W tym miejscu komenda `\string` jest wykorzystana do złożenia znaku `\` (*backslash*, poprzedzającego wszystkie komendy), znaków `{` i `}` oraz `_`. Komenda `\qqquad` składa odstęp poziomy. Pozostałe komendy w tych dwóch wierszach składają odpowiednie znaki, np. `&` składa `&`.

14: `\smallskip` jest komendą podobną do `\medskip` i `\bigskip`, umieszcza mały odstęp.

15: `\narrower` „ściska” akapit tekstu z lewej i prawej strony o wartość aktualnego wcięcia akapitu. Wcięcie jest w tym miejscu ustawione tymczasowo przy uży-

ciu komendy `\parindent` na 25 rzeczywistych punktów (czyli ignoruje się powiększenie). Lewy nawias `{` otwiera grupę ograniczającą efekt działania obu wymienionych powyżej komend. Kiedy grupa zostanie zamknięta (patrz wiersz 32), będzie przywrócone wcześniejsze ustawienie instrukcji `\parindent`.

16, 18, 20, itd.: `\item` składa wyliczenie; patrz plik wejściowy do spisu treści.

18: `\quad` jest komendą podobną do `\qqquad`, z tym że składa odstęp o połowę mniejszy.

32: Bardzo ważne jest wstawienie nawiasu zamykającego `}` po komendzie `\smallskip`. Ustawienia takie jak np. `\narrower` są wykorzystywane przez \TeX -a po przeczytaniu przez niego całego akapitu. Komenda `\smallskip` informuje program, że akapit jest skończony i można przystąpić do jego złożenia. Bez takiej komendy przed nawiasem zamykającym, \TeX opuści grupę podlegającą komendzie `\narrower` *bez pewności, że akapit się skończył*. Dyskusja o akapitach znajduje się w dodatku, zawiera listę definicji związanych z akapitami i komend kończących akapit.

§ Komendy znakowe

Plik przetwarzany w $\text{T}_{\text{E}}\text{X}$ -u jest napisany przy użyciu standardowych znaków z klawiatury komputera. Większość tych znaków będzie drukowana w pliku wynikowym w takiej postaci, w jakiej występuje na ekranie. Następujące symbole mają jednak specjalne znaczenie jako komendy $\text{T}_{\text{E}}\text{X}$ -a:

\backslash $\{$ $\}$ $\$$ $\&$ \wedge $-$ $\%$ \sim $\#$

Aby je otrzymać w takiej postaci na wydruku, należy zastosować specjalne instrukcje. Linie 7–8 w pliku wejściowym i uwagi poniżej pudełka z plikiem wejściowym pokazują, jak to zrobić. Podstawowe funkcje wymienionych powyżej symboli:

- \backslash oznacza początek wszystkich komend (z wyjątkiem niektórych wymienionych powyżej);
- $\{$ $\}$ grupują wyrażenia, informują program, którą komendę zastosować do którego fragmentu tekstu;
- $\$$ oznacza początek i koniec trybu matematycznego;
- $\&$ składa tekst, liczby itp. w kolumnach (np. w tabeli lub macierzy), oznacza skok do następnej kolumny;
- \wedge składa frakcję górną w trybie matematycznym;
- $-$ składa frakcję dolną w trybie matematycznym;
- $\%$ określa komentarz w pliku wejściowym, czyli to, co nie powinno znaleźć się w składzie; jeśli program napotka taki znak, kończy czytanie danego wiersza i przeskakuje do następnego;
- \sim oznacza pojedynczą, niełamliwą spację (w tym miejscu nie nastąpi podział wiersza);
- $\#$ oznacza numer parametru w komendzie ogólnej, program może później wstawić w to miejsce np. tekst.

Symbole te pojawiają się jeszcze w pozostałej części podręcznika. Istnieje możliwość ich przedefiniowania, tak aby inne znaki pełniły funkcję komend; zostanie to omówione w dodatku.

```

1 \noindent{\bf \S\ Kroje pisma}\hfil\break
2 \TeX\ oferuje wiele odmian krojów pisma, najczęściej używane to:\hfil\break
3 \centerline{{\rm antykwa} \quad {\sl pochyły} \quad {\it kursywa} \quad {\bf
4 pogrubiony} \quad {\tt maszynowy}}
5 Jeśli nie jest zadany jakiś inny krój, wykorzystywana jest prosta odmiana pisma
6 (antykwa). Dodatkowe kroje dostępne są w trybie matematycznym (patrz str.~15).
7 Istnieje możliwość użycia większej liczby krojów, ale tutaj zostały one
8 w większości przypadków pominięte. W przykładzie~X przedstawiono sposób
9 uzyskiwania innych krojów oraz regulacji stopnia pisma. Dłuższa lista dostępnych
10 krojów znajduje się w dodatku (pod hasłem {\sl fonty/}).
11 \bigskip
12 \noindent{\bf \S\ Odstęp}\hfil\break
13 \TeX\ określa własne odstęp między literami i słowami, wcięcie akapitu
14 i decyduje, w którym miejscu złamać wiersz czy stronę. Program czyta całe
15 akapity w pliku wejściowym, zanim wybierze odstęp zgodne z ustawionymi
16 parametrami. Jeśli zachodzi taka konieczność, można zmienić te parametry.
17 Jednostki wykorzystywane do definiowania wymiarów to m.in. {\sl cale/} (in),
18 {\sl centymetry/} (cm) i {\sl punkty/} (pt), gdzie
19 1$\,$in = 72,27$\,$pt = 2,54$\,$cm. \TeX\ rozpoznaje także inne jednostki, ale
20 pozostaniemy przy tych trzech. Na str.~{\sl vii/} określono odstęp w Plain
21 \TeX-u. Do tej pory podano już kilka przykładów zmiany tych ustawień ---
22 np. na str.~{\sl ix} i~1 --- pozostałe będą dalej. Kolejne dwie strony
23 zawierają więcej informacji na temat sterowania odstępami w tekście.
24
25 (\TeX\ reguluje wielkość odstępów w tekście, wzorach, itp., tworząc poziome
26 ({\it hbox/}) i pionowe pudełka ({\it vbox/}), a następnie łącząc je za pomocą
27 niewidocznych połączeń zwanych klejem ({\it glue/}). Sposób budowania pudełek
28 przedstawiono w przykładzie~XV; problem pudełek i kleju poruszony został
29 także w dodatku.)
30 \vfil\eject

```

UWAGI

Prawie wszystkie komendy znajdujące się na tej stronie omówiono już wcześniej. Np. wiązania ~ (*ties*).

1, 12: `\noindent` nie wcina akapitu, `\S` składa znak §, a `\hfil\break` wypełnia wiersz justunkiem aż do prawego marginesu i dokonuje podziału wiersza.

3-4: `\centerline` wyśrodkowuje materiał w danym wierszu. Komendy `\rm`, `\sl`, `\it`, `\bf` i `\tt` określają odmiany krojów pisma, odpowiednio, antykwa, *pochyły*, *kursywa*, **pogrubiony** i **maszynowy**. Domyślnym krojem pisma dla Plain \TeX -a jest antykwa. `\quad` jest komendą umieszczającą w tekście dodatkowy odstęp.

11: `\bigskip` składa odstęp pionowy.

24: Pusty wiersz w tekście informuje o początku nowego akapitu. \TeX czyta cały akapit, zanim zadecy-

duje, jak złożyć tekst; należy pamiętać, że pewne komendy, takie jak `\looseness`, odnoszą się tylko do danego akapitu. Ponieważ \TeX składa tekst akapitami a nie wierszami, stąd istnieje wiele komend służących do kształtowania akapitów. Kilka z nich zaprezentowano w przykładach I i VIII. Pełna dyskusja dotycząca akapitów znajduje się w dodatku.

30: `\vfil` wypełnia pozostałą część kolumny justunkiem, a `\eject` kończy bieżący akapit i łamie kolumnę. Aby przełamać kolumnę w środku akapitu, należy zastosować polecenie `\vadjust{\vfil\eject}`. Komenda ta spowoduje wypełnienie bieżącego wiersza justunkiem i przełamanie kolumny tekstu, a akapit będzie kontynuowany na następnej stronie.

§ Kroje pisma

\TeX oferuje wiele odmian krojów pisma, najczęściej używane to:

antykwą *pochyły* *kursywa* **pogrubiony** maszynowy

Jeśli nie jest zadany jakiś inny krój, wykorzystywana jest prosta odmiana pisma (antykwą). Dodatkowe kroje dostępne są w trybie matematycznym (patrz str. 15). Istnieje możliwość użycia większej liczby krojów, ale tutaj zostały one w większości przypadków pominięte. W przykładzie X przedstawiono sposób uzyskiwania innych krojów oraz regulacji stopnia pisma. Dłuższa lista dostępnych krojów znajduje się w dodatku (pod hasłem *fonty*).

§ Odstępy

\TeX określa własne odstępy między literami i słowami, wcięcie akapitu i decyduje, w którym miejscu złamać wiersz czy stronę. Program czyta całe akapity w pliku wejściowym, zanim wybierze odstępy zgodne z ustawionymi parametrami. Jeśli zachodzi taka konieczność, można zmienić te parametry. Jednostki wykorzystywane do definiowania wymiarów to m.in. *cale* (in), *centymetry* (cm) i *punkty* (pt), gdzie $1 \text{ in} = 72,27 \text{ pt} = 2,54 \text{ cm}$. \TeX rozpoznaje także inne jednostki, ale pozostanmy przy tych trzech. Na str. vii określono odstępy w Plain \TeX -u. Do tej pory podano już kilka przykładów zmiany tych ustawień — np. na str. ix i 1 — pozostałe będą dalej. Kolejne dwie strony zawierają więcej informacji na temat sterowania odstępami w tekście.

(\TeX reguluje wielkość odstępów w tekście, wzorach, itp., tworząc poziome (*hbox*) i pionowe pudełka (*vbox*), a następnie łącząc je za pomocą niewidocznych połączeń zwanych klejem (*glue*). Sposób budowania pudełek przedstawiono w przykładzie XV; problem pudełek i kleju poruszony został także w dodatku.)


```

1 \textindent{ $\bullet$ } \underbar{\sl ODSTĘPY MIĘDZY WYRAZAMI}. Podczas
2 przetwarzania tekstu (inaczej niż w trybie matematycznym), \TeX\ interpretuje
3 jedną lub więcej spacji jako {\sl pojedynczy\/} odstęp międzywyrazowy. Nie
4 musimy się więc martwić o~odstępy między wyrazami w~pliku wejściowym
5 (patrz wiersz 4 w~ramce na stronie obok). W razie potrzeby możemy uzyskać
6 poziomy odstęp dowolnej długości (patrz wiersz~8 w~pliku
7 wejściowym):\hfil\break
8 \centerline{[], [\negthinspace], [\thinspace], [\ ], [\quad], [\qquad].}
9 Pierwszy jest „normalnym” międzyznakowym odstępem, następny małą spacją
10 „ujemną”, pozostałe odstępami dodatnimi różnej długości. Komendy tworzące
11 ostatnie trzy odstępy (znak {\tt\string\ }i odstęp, {\tt\string\quad}
12 i~{\tt\string\qquad}) są często używane do regulacji odległości. Odstęp poziomy
13 dowolnej długości  $\$1\$$  można uzyskać za pomocą wpisu „{\tt\string\hskip}  $\$1\$$ ”,
14 gdzie  $\$1\$$  musi być podane w~jednostkach rozpoznawanych przez program. \TeX\
15 automatycznie wstawia spację pomiędzy ostatnim wyrazem w~wierszu pliku
16 wejściowego a pierwszym wyrazem wiersza następnego. Aby zlikwidować ten odst%
17 ęp, należy wstawić w danym miejscu znak {\tt \%} (patrz ramka na stronie obok,
18 wiersz 16).
19
20 Ponieważ \TeX\ formatuje tekst wyrównując go do prawego marginesu, odstępy
21 między wyrazami muszą być elastyczne. Niemniej jednak określona jest minimalna
22 i~maksymalna odległość, w jakiej mogą pozostawać dwa wyrazy. Odstępy można zadać
23 w sposób elastyczny, np. komenda „{\tt\string\hskip 10pt plus 2pt minus 3pt}”
24 składa 10-punktowy odstęp, który może się rozciągnąć o~2~punkty (i~więcej, jeśli
25 to jest konieczne) lub ścisnąć o~3~punkty.
26 \medskip\textindent{ $\bullet$ }
27 \underbar{\sl WIERSZE}. \TeX\ decyduje, kiedy dokonać podziału wiersza.
28 Jeśli nie ma podanych innych instrukcji, korzysta z~ustawień dotyczących
29 długości wiersza i~interlinii (na str.~1 były zastosowane takie instrukcje).
30 Podobnie jak odstępy poziome, można uzyskać odstępy pionowe o zadanych
31 parametrach. Komendy tworzące takie odstępy to m.in. {\tt\string\smallskip},
32 {\tt\string\medskip} i~{\tt\string\bigskip}. Efekt działania tych komend
33 przedstawiają odległości pomiędzy kreskami na następnej stronie.
34 \vfil\eject

```

UWAGI

1: `\textindent{symbol}` wcina tekst początku akapitu i umieszcza symbol we wcięciu. `\bullet` składa znak •. `\underbar` podkreśla tekst. `\/` oznacza poprawkę do pisma pochylego (korektę italikową), tzn. mały odstęp zaraz po pochylej odmianie pisma lub kursywie.

4: Na stronie składu można zobaczyć, jak kolejne odstępy istniejące w pliku wejściowym tego wiersza są ignorowane przez program. \TeX zostawia trochę większy odstęp, niż normalny międzywyrazowy, po niektórych znakach interpunkcji (np. po kropce).

8: Komenda `\negthinspace` składa mały „ujemny” odstęp, nieznacznie cofa tekst. Symbol „æ” otrzymamy (aczkolwiek paskudnie to wygląda) za pomocą

komendy „`a \negthinspace e`”. `\thinspace` składa wąski odstęp, np. „DiStefano” uzyskano z wpisu „`Di\thinspace Stefano`”. Umieszczenie w pliku wejściowym znaku `\` ze spacją skutkuje w składzie odstępem równym zwykłej spacji międzywyrazowej. Komendy `\quad` i `\qquad` składają większe odstępy.

12, 13, 23, itd.: Komenda `\string` umożliwia skład dosłowny następującego po niej tekstu. `\tt` składa tekst maszynowym krojem pisma.

26: `\medskip` składa odstęp pionowy. Zwróć uwagę na całkowitą dowolność, na jaką \TeX pozwala przy umieszczaniu takich komend — w osobnym wierszu lub w tym samym wierszu, nad lub pod tekstem.

34: `\vfil\eject` omówiono na str. 8.

- ODSTĘPY MIĘDZY WYRAZAMI. Podczas przetwarzania tekstu (inaczej niż w trybie matematycznym), \TeX interpretuje jedną lub więcej spacji jako *pojedynczy* odstęp międzywyrazowy. Nie musimy się więc martwić o odstępy między wyrazami w pliku wejściowym (patrz wiersz 4 w ramce na stronie obok). W razie potrzeby możemy uzyskać poziomy odstęp dowolnej długości (patrz wiersz 8 w pliku wejściowym):

$$[], [], [], [], [], [\quad].$$

Pierwszy jest „normalnym” międzyznakowym odstępem, następny małą spacją „ujemną”, pozostałe odstępami dodatnimi różnej długości. Komendy tworzące ostatnie trzy odstępy (znak \backslash i odstęp, \backslashquad i \backslashqqquad) są często używane do regulacji odległości. Odstęp poziomy dowolnej długości l można uzyskać za pomocą wpisu „ $\backslashhskip l$ ”, gdzie l musi być podane w jednostkach rozpoznawanych przez program. \TeX automatycznie wstawia spację pomiędzy ostatnim wyrazem w wierszu pliku wejściowego a pierwszym wyrazem wiersza następnego. Aby zlikwidować ten odstęp, należy wstawić w danym miejscu znak % (patrz ramka na stronie obok, wiersz 16).

Ponieważ \TeX formatuje tekst wyrównując go do prawego marginesu, odstępy między wyrazami muszą być elastyczne. Niemniej jednak określona jest minimalna i maksymalna odległość, w jakiej mogą pozostawać dwa wyrazy. Odstępy można zadać w sposób elastyczny, np. komenda „ $\backslashhskip10pt plus 2pt minus 3pt$ ” składa 10-punktowy odstęp, który może się rozciągnąć o 2 punkty (i więcej, jeśli to jest konieczne) lub ścisnąć o 3 punkty.

- WIERSZE. \TeX decyduje, kiedy dokonać podziału wiersza. Jeśli nie ma podanych innych instrukcji, korzysta z ustawień dotyczących długości wiersza i interlinii (na str. 1 były zastosowane takie instrukcje). Podobnie jak odstępy poziome, można uzyskać odstępy pionowe o zadanych parametrach. Komendy tworzące takie odstępy to m.in. \backslashsmallskip , \backslashmedskip i \backslashbigskip . Efekt działania tych komend przedstawiają odległości pomiędzy kreskami na następnej stronie.

```

1 \hrule width5cm\hfil\break
2 \line{\hfill}
3 \hrule width5cm\smallskip
4 \hrule width5cm\medskip
5 \hrule width5cm\bigskip
6 \hrule width5cm
7 \line{\hfill}\smallskip
8 \noindent Możliwe jest także złożenie odstępu pionowego o~określonej wysokości
9 $h$, za pomocą komendy {\tt \string \vskip} $h$ ($h$ musi być podane
10 w~jednostkach rozpoznawanych przez program). Podział wiersza jest wymuszony
11 działaniem zestawów komend {\tt\string\hfil\string\break} lub
12 {\tt\string\hfill\string\break}. {\tt\string\break} informuje o~podziale
13 wiersza, komendy poprzedzające wypełniają wiersz justunkiem. Spróbujmy opuścić
14 komendę {\tt\string\hfil} i~zobaczyć, co się stanie. Z~dwóch komend
15 ,,wypełniających'', {\tt\string\hfill} jest silniejsza niż {\tt\string\hfil}
16 i ma pierwszeństwo przed wcześniej wprowadzonymi komendami dotyczącymi odstępow.
17 Przykład: eksponowane wyrażenia matematyczne są w \TeX-u automatycznie
18 wyśrodkowane, mogą jednak być dosunięte do prawego (lub lewego) marginesu przez
19 zastosowanie komendy {\tt\string\hfill} z~lewej lub prawej strony równania ---
20 komenda {\tt\string\hfil} nie zadziała w~tym miejscu (patrz plik wejściowy
21 równania~6 w~przykładzie~IX).
22 \medskip\textindent{${\bullet}$}
23 \underbar{\sl AKAPITY}. Pusty wiersz jest znakiem końca akapitu. W~tym samym
24 celu może być wykorzystana komenda {\tt\string\par} (patrz plik wejściowy,
25 linia~25).\par Akapity są automatycznie wcinane i odsuwane na określoną
26 odległość od przetwarzanego tekstu. Ustalenia te mogą być zmieniane za pomocą
27 stosownych komend (patrz pliki wejściowe str.~{\sl ix\} i~1).
28 \medskip\textindent{${\bullet}$}
29 \underbar{\sl STRONY}. \TeX\ sam decyduje, w którym miejscu zakończyć stronę
30 i~rozpocząć nową. Łamanie kolumny tekstu może zostać wymuszone podobnie,
31 jak podział wierszy (patrz na koniec pliku wejściowego tej strony). Wysokość
32 kolumny kontrolowana jest przez ,,rozmiar pionowy'' ({\it vertical size\}),
33 ustalony na początku pliku wejściowego str.~1.
34 \medskip
35 Więcej informacji na temat odstępow znajduje się w dodatku.
36 \vfil\ejct

```

UWAGI

1, 3–6: `\hrule` składa kreskę (rule), tzn. prostokąt wypełniony czarną farbą drukarską. Kreska ta ma wysokość, szerokość i głębokość (w drukarstwie nazywaną odsadką). Domyślną wartością odsadki jest `Opt`, wysokości `0,4pt`, szerokości — rzeczywista szerokość tekstu (szerokość kolumny lub szerokość innej struktury, np. tabeli, wewnątrz której kreska się znajduje). Określenie któregośkolwiek z tych parametrów powoduje zmianę ustawień domyślnych.

2, 7: `\line{\hfill}` składa pusty wiersz, wypełniony justunkiem. Komenda `\hfill` działa podobnie do `\hfil`, jest jednak silniejsza, tzn. ma pierwszeń-

stwo przed innymi komendami dotyczącymi odstępow, włączając `\hfil`. W tym miejscu akurat nie była potrzebna, ale nie robi tu żadnej szkody, a jest odpowiedni czas na jej przedstawienie.

25: `\par` kończy akapit. Jeśli akapit jest zakończony pustą linijką w tekście, `\TeX` zamienia ją na komendę `\par`, którą również wstawia, gdy w trybie poziomym (podczas czytania akapitu) napotka komendę z trybu pionowego, np. `\vfil`. Jest to przydatne w niektórych sytuacjach, np. można umieszczać dowolny obiekt/znak na końcu każdego z akapitów, definiując komendę `\par` (przykład XIII).

Możliwe jest także złożenie odstępów pionowych o określonej wysokości h , za pomocą komendy `\vskip h` (h musi być podane w jednostkach rozpoznawanych przez program). Podział wiersza jest wymuszony działaniem zestawów komend `\hfil\break` lub `\hfill\break`. `\break` informuje o podziale wiersza, komendy poprzedzające wypełniają wiersz justunkiem. Spróbujmy opuścić komendę `\hfil` i zobaczyć, co się stanie. Z dwóch komend „wypełniających”, `\hfill` jest silniejsza niż `\hfil` i ma pierwszeństwo przed wcześniej wprowadzonymi komendami dotyczącymi odstępów. Przykład: eksponowane wyrażenia matematyczne są w \TeX -u automatycznie wyśrodkowane, mogą jednak być dosunięte do prawego (lub lewego) marginesu przez zastosowanie komendy `\hfill` z lewej lub prawej strony równania — komenda `\hfil` nie zadziała w tym miejscu (patrz plik wejściowy równania 6 w przykładzie IX).

- AKAPITY. Pusty wiersz jest znakiem końca akapitu. W tym samym celu może być wykorzystana komenda `\par` (patrz plik wejściowy, linia 25).

Akapity są automatycznie wcinane i odsuwane na określoną odległość od przetwarzanego tekstu. Ustalenia te mogą być zmieniane za pomocą stosownych komend (patrz pliki wejściowe str. ix i 1).

- STRONY. \TeX sam decyduje, w którym miejscu zakończyć stronę i rozpocząć nową. Łamanie kolumny tekstu może zostać wymuszone podobnie, jak podział wierszy (patrz na koniec pliku wejściowego tej strony). Wysokość kolumny kontrolowana jest przez „rozmiar pionowy” (*vertical size*), ustalony na początku pliku wejściowego str. 1.

Więcej informacji na temat odstępów znajduje się w dodatku.

```

1 \noindent{\bf \S Tryb matematyczny}\hfil\break
2 Wzory matematyczne w tekście muszą być ograniczone znakami {\tt \$}, a wzory
3 eksponowane (wyśrodkowane w osobnym wierszu) --- dwoma znakami {\tt \$\$}.
4 \smallskip\textindent{\$\bullet\$}
5 \underbar{\sl KRÓJ PISMA}. Standardowy krój pisma w tym trybie pracy
6 ( $\textit{kursywa}$   $\text{\$matematyczna}$ ) jest jedynie podobny do  $\textit{kursywy tekstowej}$ .
7 Czy jesteś w stanie zauważyć różnicę? Tekst równań matematycznych można drukować
8 także innym krojem; patrz przykład IX, równania 6, 9, 12 i 13. Inne dostępne
9 kroje pisma to  $\text{\cal KALIGRAFICZNY}$  (tylko duże litery) i tzw.
10  $\text{\scriptstyle Script}$  oraz  $\text{\scriptscriptstyle Scriptscript}$  (wykorzystywane
11 do indeksów i innych wyrażeń „drobnym drukiem”). Krój normalnego rozmiaru
12 jest nazywany stylem tekstowym, a krój większego stopnia, stosowany do
13 eksponowanych równań --- stylem eksponowanym. Część wzoru może być złożona
14 innym krojem niż automatycznie wprowadzany przez program (patrz przykład V,
15 równanie 1c). Dostępne są litery greckie ( $\phi$ ,  $\epsilon$ ,  $\gamma$ ,
16  $\Gamma$ , itd.) oraz mnóstwo symboli specjalnych (np.  $\cup$ ,  $\bigcup$ ,
17  $\infty$ ,  $\sum$ ,  $\emptyset$ ,  $\in$ ); ich wykaz znajduje się w dodatku.
18 \smallskip\textindent{\$\bullet\$}
19 \underbar{\sl ODSTĘPY}. Zasady tworzenia odstępów w trybie matematycznym są
20 trochę inne niż w trybie tekstowym, stąd nie nadają się do składu tekstu:
21 np.  $\text{\$sufferi n g}$ . Program ignoruje puste odstępy w pliku wejściowym
22 (patrz plik wejściowy, wiersz 21), co pozwala na dowolne aranżowanie pliku
23 wejściowego, a nie wpływa na odstępy w składzie. Odstępy poziome można
24 regulować tak jak pokazano poniżej w nawiasach:
25  $[\ ] , \ ; [ \ ] ! , \ ; [ \ ] , \ ; [ \ ] > , \ ; [ \ ] ; , \ ; [ \ ] \quad , \ ; [ \ ] \quad \quad . \$ \$$ 
26 Pierwszy z odstępów to normalna spacja, drugi „ujemna wąska spacja”,
27 a pozostałe to odstępy dodatnie. Odstępy w wyrażeniach matematycznych omówiono
28 szerzej w dodatku. Eksponowany wzór matematyczny  $\text{\TeX}$  stara się umieścić
29 w jednym wierszu, tak więc w przypadku długich wyrażeń dobrze jest ręcznie
30 podzielić je w najodpowiedniejszym miejscu. Wielowierszowe układy wyrażeń
31 matematycznych mogą być eksponowane na wiele sposobów (przykłady V, VII i IX).
32 \vfil\eject

```

UWAGI

4, 6, itd.: Znak $\text{\$}$, omawiany już, włącza i wyłącza tryb matematyczny.

9, 10: \cal , \scriptstyle , itd. informują o zastosowaniu specjalnych krojów pisma w trybie matematycznym, patrz strona sąsiednia.

15–17: Efekt działania komend \phi , ..., \in można zauważyć patrząc na sąsiednią stronę. Oddzielne znaki $\text{\$}$ nie są konieczne dla każdej z nich (zapis \phi , \epsilon , \gamma \Gamma jest również prawidłowy), preferuje się jednak zastosowanie osobnych par do każdego wyrażenia matematycznego. Patrz dyskusja w dodatku pod hasłem matematyka: interpunkcja.

25: \! , \, , \> , \; są komendami odstępów. Efekty

ich działania widoczne są na stronie składu. Znaki $\text{\$ \$}$ oznaczają początek i koniec eksponowanego trybu matematycznego. Eksponowane wyrażenia matematyczne są umieszczane w osobnym wierszu, automatycznie wyśrodkowane, z dodatkowym odstępem nad i pod wyrażeniem.

KOMENTARZ

\TeX nie potrafi dzielić wzorów matematycznych; eksponowane wzory nigdy nie są automatycznie dzielone na dwa wiersze, należy ręcznie określić miejsca podziału. W dodatku znajduje się omówienie takich problemów pod hasłem matematyka: podział wierszy.

§ Tryb matematyczny

Wzory matematyczne w tekście muszą być ograniczone znakami \$, a wzory eksponowane (wyśrodkowane w osobnym wierszu) — dwoma znakami \$.

- KRÓJ PISMA. Standardowy krój pisma w tym trybie pracy (*kursywa matematyczna*) jest jedynie podobny do *kursywy tekstowej*. Czy jesteś w stanie zauważyć różnicę? Tekst równań matematycznych można drukować także innym krojem; patrz przykład IX, równania 6, 9, 12 i 13. Inne dostępne kroje pisma to *KALIGRAFICZNY* (tylko duże litery) i tzw. *Script* oraz *Scriptscript* (wykorzystywane do indeksów i innych wyrażeń „drobnym drukiem”). Krój normalnego rozmiaru jest nazywany stylem tekstowym, a krój większego stopnia, stosowany do eksponowanych równań — stylem eksponowanym. Część wzoru może być złożona innym krojem niż automatycznie wprowadzany przez program (patrz przykład V, równanie 1c). Dostępne są litery greckie (ϕ , ϵ , γ , Γ , itd.) oraz mnóstwo symboli specjalnych (np. \cup , \bigcup , ∞ , \sum , \emptyset , ϵ); ich wykaz znajduje się w dodatku.

- ODSTĘPY. Zasady tworzenia odstępów w trybie matematycznym są trochę inne niż w trybie tekstowym, stąd nie nadają się do składu tekstu: np. *suffering*. Program ignoruje puste odstępów w pliku wejściowym (patrz plik wejściowy, wiersz 21), co pozwala na dowolne aranżowanie pliku wejściowego, a nie wpływa na odstępów w składzie. Odstępów poziome można regulować tak jak pokazano poniżej w nawiasach:

$$[], [], [], [], [], [], [].$$

Pierwszy z odstępów to normalna spacja, drugi „ujemna wąska spacja”, a pozostałe to odstępów dodatnie. Odstępów w wyrażeniach matematycznych omówiono szerzej w dodatku. Eksponowany wzór matematyczny T_EX stara się umieścić w jednym wierszu, tak więc w przypadku długich wyrażeń dobrze jest ręcznie podzielić je w najodpowiedniejszym miejscu. Wielowierszowe układy wyrażeń matematycznych mogą być eksponowane na wiele sposobów (przykłady V, VII i IX).

```

1 \noindent{\bf \S\ Przetwarzanie pliku wejściowego}\hfil\break
2 Plik wejściowy dla \TeX-a powinien mieć nazwę typu {\it nazwa\_pliku}.tex
3 i~powinien być zakończony słówkiem {\tt\string\end} lub {\tt\string\bye}
4 (komenda ta została omówiona na str.~{\sl xii\} oraz w~dodatku). Jeśli plik
5 nie kończy się w~ten sposób, po przetworzeniu go program wyświetli znak $\ast$
6 i~będzie oczekiwał na reakcję w~postaci komendy {\tt\string\end}.
7
8 Aby \TeX\ przetworzył plik, należy wpisać np. {\tt tex} {\it nazwa\_pliku\}.
9 Komenda ta może różnić się w~różnych systemach, dlatego trzeba dowiedzieć się
10 u~lokalnego operatora, jaka jest odpowiednia dla systemu, w~którym pracujemy.
11 Plik jest przetwarzany strona po stronie; jeśli program znajdzie jakiś błąd, od
12 razu wyświetli odpowiedni komunikat. Błędy omówiono skrótowo poniżej,
13 a~szerzej --- w~dodatku. Gotowe strony \TeX\ umieszcza w~pliku o~nazwie {\it
14 nazwa\_pliku}.dvi, gdzie {\sl dvi\} oznacza „niezależny od
15 urządzenia” ({\it device independent\}). Plik {\sl dvi\} zawiera jawne
16 instrukcje zrozumiałe dla urządzenia drukującego (patrz także dodatek,
17 hasło {\sl dvi\}).
18
19 Utworzony plik {\sl dvi\} może zostać wydrukowany, należy tylko wiedzieć,
20 jakiej komendy {\tt drukowania} użyć w~danym systemie.
21 \bigskip
22 \noindent{\bf \S\ Błędy}\hfil\break
23 Jeśli \TeX\ wyświetla nam wiadomość o~błędzie, wciskamy klawisz {\tt h}, aby
24 otrzymać dodatkowe informacje, a~następnie {\tt e} --- w~celu ponownej edycji
25 pliku. Jeśli błędy mają zasięg lokalny i~nie wywierają wpływu na dalszą część
26 tekstu, lepiej jest przetwarzać plik w~całości. Program zatrzymuje się przy
27 każdym z~błędów, można więc je wynotować, wciskając klawisz {\tt enter} (lub
28 {\tt return}) w~celu kontynuacji przetwarzania, a~następnie wyedytować plik
29 i~dokonać odpowiedniej korekty. Jeśli wykonujemy jedynie prosty skład,
30 informacje dotyczące błędów (jeśli będą takowe) będą prawdopodobnie
31 zrozumiałe, np. brak znaku {\tt$}, nawiasu otwierającego lub
32 zamykającego ({\tt\string{ }~i~{\tt\string}}) lub przekreślona nazwa komendy.
33
34 Najczęściej spotykane komendy dotyczące błędów są omówione w~dodatku
35 (pod hasłem {\sl błędy\}).
36 \vfil\ejct

```

§ Przetwarzanie pliku wejściowego

Plik wejściowy dla T_EX-a powinien mieć nazwę typu *nazwa_pliku.tex* i powinien być zakończony słówkiem `\end` lub `\bye` (komenda ta została omówiona na str. *xii* oraz w dodatku). Jeśli plik nie kończy się w ten sposób, po przetworzeniu go program wyświetli znak `*` i będzie oczekiwał na reakcję w postaci komendy `\end`.

Aby T_EX przetworzył plik, należy wpisać np. `tex nazwa_pliku`. Komenda ta może różnić się w różnych systemach, dlatego trzeba dowiedzieć się u lokalnego operatora, jaka jest odpowiednia dla systemu, w którym pracujemy. Plik jest przetwarzany strona po stronie; jeśli program znajdzie jakiś błąd, od razu wyświetli odpowiedni komunikat. Błędy omówiono skrótowo poniżej, a szerzej — w dodatku. Gotowe strony T_EX umieszcza w pliku o nazwie *nazwa_pliku.dvi*, gdzie *dvi* oznacza „niezależny od urządzenia” (*device independent*). Plik *dvi* zawiera jawne instrukcje zrozumiałe dla urządzenia drukującego (patrz także dodatek, hasło *dvi*).

Utworzony plik *dvi* może zostać wydrukowany, należy tylko wiedzieć, jakiej komendy **drukowania** użyć w danym systemie.

§ Błędy

Jeśli T_EX wyświetla nam wiadomość o błędzie, wciskamy klawisz `h`, aby otrzymać dodatkowe informacje, a następnie `e` — w celu ponownej edycji pliku. Jeśli błędy mają zasięg lokalny i nie wywierają wpływu na dalszą część tekstu, lepiej jest przetwarzać plik w całości. Program zatrzymuje się przy każdym z błędów, można więc je wynotować, wciskając klawisz `enter` (lub `return`) w celu kontynuacji przetwarzania, a następnie wyedytować plik i dokonać odpowiedniej korekty. Jeśli wykonujemy jedynie prosty skład, informacje dotyczące błędów (jeśli będą takowe) będą prawdopodobnie zrozumiałe, np. brak znaku `$`, nawiasu otwierającego lub zamykającego (`{ i }`) lub przekreślona nazwa komendy.

Najczęściej spotykane komendy dotyczące błędów są omówione w dodatku (pod hasłem *błędy*).


```

1 \noindent{\bf \S\ Uwagi ogólne}\hfil\break
2 Wstępne uwagi z ostatnich kilku stron są jedynie szkicowe, bardziej kompletną
3 listę zasad i komend umieszczono w dodatku. Na początku możliwe jest
4 popełnianie pomyłek, w wyniku ćwiczeń dochodzi się do wprawy. W przykładach
5 w kolejnych podrozdziałach pokazano, co można zrobić za pomocą TeX-a.
6 W przypadku korzystania z wersji zawierającej bardziej zaawansowane komendy
7 niż te z Plain TeX-a, nie wszystko może działać dokładnie tak, jak to
8 przedstawiono w tym podręczniku. Wówczas trzeba trochę poeksperymentować.
9
10 Jest wiele różnych sposobów formatowania, tutaj ograniczymy się tylko do kilku
11 podstawowych reguł i w zasadzie nie będą stosowane inne warianty. {\it Aby plik
12 wejściowy był czytelny, zostaną pominięte różne małe subtelnosci, do jakich
13 TeX\ jest zdolny (np. dopasowanie odstępów), jeśli będzie to przeszkadzać
14 wyjaśnieniu głównych przedstawianych spraw}. W rezultacie strony wynikowe
15 nie zawsze będą tak eleganckie, jak mogłyby być, ale powinny przedstawić
16 niektóre możliwości programu.
17
18 {Na początku podręcznika zrezygnowano z definiowania nowych komend, nawet
19 jeśli mogły one ułatwić pracę przy składzie tej książki. Dopiero w dalszych
20 przykładach i w dodatku skorzystano w pełni z tych udogodnień. Komendy tam
21 przedstawione powinny pokazać użytkownikowi, jak można utworzyć własne
22 środowisko ({\it framework\}), w którym praktycznie wszystkie operacje
23 formatujące są ukryte w kilku prostych instrukcjach. Środowisko takie, raz
24 zdefiniowane, pozwala skoncentrować się wyłącznie na treści dokumentu, bez
25 zakłócania go dużą liczbą komend {\tt\string\hfil} i {\tt\string\noindent}.
26 Jeśli zamierza się regularnie korzystać z TeX-a, dobrze będzie stworzyć
27 własny format składu. TeX\ jest potężnym {\it językiem}, służącym do składu
28 tekstu, i tylko w przypadku, gdy korzysta się z niego konstruując swoje własne
29 formaty, wykorzystuje się całą jego moc.
30 \parfillskip0pt\par}
31 \vfil\break

```

UWAGI

18–30: Ostatni akapit na tej stronie został ujęty w nawiasy, ponieważ zmieniono jedno z ustawień akapitu, o czym mówi następna uwaga.

30: `\parfillskip` informuje program, jak ma złożyć ostatni wiersz akapitu. Domyślnie wbudowane jest wypełnienie `\hfil` i na końcu wiersza, w razie potrzeby, umieszczany jest klej. Zmiana wartości parametru `\parfillskip` na `0pt` wymusza zakończenie akapitu z dosunięciem do prawego marginesu. Inne wartości powodują dowolne zakończenie ostatniego wiersza. `\par` jest wtedy umieszczane

w celu wyraźnego zakończenia akapitu, przed końcem grupy ograniczającej działanie lokalnego ustawienia `\parfillskip`. W niniejszym przykładzie takie ustawienie parametru powoduje mało estetyczne, nadmierne rozsuniecie wyrazów w ostatniej linii akapitu.

Warto zauważyć, że kontrolowanie ostatniego wiersza akapitu jest możliwe tylko dlatego, że T_EX składa dany akapit w całości, zamiast wiersz po wierszu.

31: `\vfil` wypełnia kolumnę justunkiem; w tym miejscu `\break` jest komendą końca strony, podobnie jak `\eject` (patrz dodatek, hasło `\break`).

§ Uwagi ogólne

Wstępne uwagi z ostatnich kilku stron są jedynie szkicowe, bardziej kompletną listę zasad i komend umieszczono w dodatku. Na początku możliwe jest popełnianie pomyłek, w wyniku ćwiczeń dochodzi się do wprawy. W przykładach w kolejnych podrozdziałach pokazano, co można zrobić za pomocą T_EX-a. W przypadku korzystania z wersji zawierającej bardziej zaawansowane komendy niż te z Plain T_EX-a, nie wszystko może działać dokładnie tak, jak to przedstawiono w tym podręczniku. Wówczas trzeba trochę poeksperymentować.

Jest wiele różnych sposobów formatowania, tutaj ograniczymy się tylko do kilku podstawowych reguł i w zasadzie nie będą stosowane inne warianty. *Aby plik wejściowy był czytelny, zostaną pominięte różne małe subtelnosci, do jakich T_EX jest zdolny (np. dopasowanie odstępów), jeśli będzie to przeszkadzać wyjaśnieniu głównych przedstawianych spraw.* W rezultacie strony wynikowe nie zawsze będą tak eleganckie, jak mogłyby być, ale powinny przedstawić niektóre możliwości programu.

Na początku podręcznika zrezygnowano z definiowania nowych komend, nawet jeśli mogły one ułatwić pracę przy składzie tej książki. Dopiero w dalszych przykładach i w dodatku skorzystano w pełni z tych udogodnień. Komendy tam przedstawione powinny pokazać użytkownikowi, jak można utworzyć własne środowisko (*framework*), w którym praktycznie wszystkie operacje formatujące są ukryte w kilku prostych instrukcjach. Środowisko takie, raz zdefiniowane, pozwala skoncentrować się wyłącznie na treści dokumentu, bez zakłócania go dużą liczbą komend `\hf il` i `\noindent`. Jeśli zamierza się regularnie korzystać z T_EX-a, dobrze będzie stworzyć własny format składu. T_EX jest potężnym *językiem*, służącym do składu tekstu, i tylko w przypadku, gdy korzysta się z niego konstruując swoje własne formaty, wykorzystuje się całą jego

moc.

```

1 \line{\hfil\title Przykłady}% \title NIE JEST STANDARDOWĄ KOMENDĄ
2 \headline{\hfil}
3 \vskip 3,3 true in
4 \centerline{\bf I. PROSTO O PIŚMIENICTWIE}
5 \centerline{\sl Eksponowanie tekstu w wąskim lub szerokim formacie;
6 modelowanie akapitów.}
7 \bigskip
8 \noindent Opracowanie dotyczące upadku języka angielskiego, opublikowane w 1946
9 roku2, zawiera ustęp z Eklezjastesu, przetłumaczony przez George Orwella na
10 „nowoczesny angielski najgorszego rodzaju”. Oto oryginalny fragment:
11 \smallskip
12 {\narrower \noindent
13 \sl Odwróciłem się i zobaczyłem w słońcu, że bieg nie jest dla szybkiego ani
14 walka dla silnego, ani również chleb dla mądrego, ani nawet bogactwa dla
15 rozumiejącego człowieka, ani wyrozumiałość dla zręcznego człowieka; jednak czas
16 i możliwości zdarzają się im wszystkim.\smallskip}
17 \noindent A oto „tłumaczenie” Orwella:\smallskip
18 {\leftskip=1cm \rightskip=,8cm \noindent
19 \it Obiektywne rozważenie współczesnych zjawisk prowadzi do wniosku, że sukces
20 lub upadek konkurencyjnych działalności nie wykazuje tendencji, by być
21 proporcjonalnym do wrodzonych zdolności, ale ten ważny element
22 nieprzewidywalności powinien być niezmiennie brany pod uwagę. \smallskip}
23 \noindent Orwell twierdził, że chociaż jego tłumaczenie było parodią, sposób
24 pisania, jaki w nim zaprezentował, to „osiągnięcie dna w nowoczesnym
25 angielskim”.
```

UWAGI

1: `\line` umieszcza dany materiał w jednym wierszu. `\hfil` wypełnia wiersz justunkiem. `\title` jest własną komendą, zdefiniowaną na str. 1. *Nie będzie działać bez uprzedniego zdefiniowania.* % to znak komentarza: TeX ignoruje tekst następujący po tym znaku.

2: `\headline` składa nagłówek. Tutaj nagłówek jest wypełniony justunkiem w celu skasowania ustawienia przeniesionego z poprzedniej strony.

3: `\vskip` pozwala na skok w pionie.

4, 5: `\centerline` wyśrodkowuje tekst. Komendy `\bf` i `\sl` wybierają, odpowiednio, **pogrubiony** i *pochyły* krój pisma.

7: `\bigskip` składa odstęp pionowy.

8, 12, itd.: `\noindent` likwiduje wcięcie akapitu.

9: `2` składa 2 jako indeks górny. Komenda ta działa wyłącznie w trybie matematycznym; znaki `$` włączają i wyłączają ten tryb.

10, itd.: Zauważ, jak składa się polskie cudzysłowy.

11, 17, itd.: `\smallskip` wstawia mały odstęp pionowy.

12: `\narrower` zwięża tekst obustronnie o odległość równą wcięciu akapitu. Wewnątrz nowych marginesów akapity są składane jak wcześniej: pierwszy wiersz jest automatycznie wcięty, itd. Komenda `\noindent`

likwiduje wcięcie, nawet jeśli jest umieszczona w linii nad akapitem.

13: `\sl` składa tekst pochyłą odmianą kroju pisma.

16: Akapit jest zakończony komendą `\smallskip`, nawias zamykający `}` określa obszar działania komend `\narrower` i `\sl`.

18: `\leftskip` i `\rightskip` pozwalają na kontrolę nad marginesami akapitu. Wartości dodatnie tych parametrów pozwalają na skład taki, jak w tym miejscu. Na następnej stronie przedstawiono efekt, jaki powodują ujemne wartości tych parametrów.

19: `\it` składa tekst kursywą.

22: Koniec akapitu musi być wyraźnie zaznaczony (tu za pomocą komendy `\smallskip`), zanim grupa określająca nowe marginesy zostanie zamknięta (znakiem `}`). Niektóre parametry formatujące akapit przyjmują wartości aktualne w momencie jego zakończenia. Jeżeli grupa ograniczająca działanie nowych parametrów formatujących zostanie zamknięta przed wyraźnie określonym końcem akapitu — zmiany te nie będą uwzględnione. Akapit zostanie sformatowany zgodnie z parametrami obowiązującymi przed otwarciem grupy. Więcej informacji na ten temat znajduje się w dodatku pod hasłem **akapity**.

Przykłady

I. PROSTO O PIŚMIENNICTWIE

Ekspozycja tekstu w wąskim lub szerokim formacie; modelowanie akapitów.

Opracowanie dotyczące upadku języka angielskiego, opublikowane w 1946 roku², zawiera ustęp z Eklezjastesu, przetłumaczony przez George Orwella na „nowoczesny angielski najgorszego rodzaju”. Oto oryginalny fragment:

Odwróciłem się i zobaczyłem w słońcu, że bieg nie jest dla szybkiego ani walka dla silnego, ani również chleb dla mądrego, ani nawet bogactwa dla rozumiejącego człowieka, ani wyrozumiałość dla zręcznego człowieka; jednak czas i możliwości zdarzają się im wszystkim.

A oto „tłumaczenie” Orwella:

Obiektywne rozważenie współczesnych zjawisk prowadzi do wniosku, że sukces lub upadek konkurencyjnych działalności nie wykazuje tendencji, by być proporcjonalnym do wrodzonych zdolności, ale ten ważny element nieprzewidywalności powinien być niezmiennie brany pod uwagę.

Orwell twierdził, że chociaż jego tłumaczenie było parodią, sposób pisania, jaki w nim zaprezentował, to „osiągnięcie dna w nowoczesnym angielskim”.

```

1 Miał rację. A~oto kilka początkowych wierszy z~wielkiej~powieści$~3$:~\smallskip
2 {\leftskip=-,5cm\rightskip=-,5cm\noindent {\sl Dziesięć tygodni przed śmiercią,
3 Mohun Biswas, dziennikarz z~Sikkim Street, St James, Port of Spain, został
4 zwolniony z~pracy. Chorował od jakiegoś czasu. Przed niespełna rokiem spędził
5 ponad 9~tygodni w~szpitalu Colonial, potem dochodził długo do zdrowia w~domu.
6 Kiedy lekarz zalecił mu całkowity wypoczynek, Trinidad Sentinel nie miało
7 wyboru. Pan~Biswas dostał 3-miesięczne wypowiedzenie, ale aż do śmierci był
8 zaopatrywany każdego ranka w darmowy egzemplarz gazety.}\smallskip}
9 \noindent I~bardziej typowy rodzaj piśmiennictwa, jaki spotykamy dzisiaj
10 (szczególnie w czasopismach naukowych) --- początkowy fragment eseju napisanego
11 przez krytyka literackiego, dotyczący wyżej cytowanej powieści$~4$:~\smallskip
12 \parshape=5 2,64true in ,385true in 1,66true in 2,3true in 1,35true in 2,8true
13 in ,5true in 4,5true in ,9true in 3,71true in
14 {\it\parindent=0pt Ktoś, czytając powieść V.S.~Naipaula {\sl Dom dla Pana
15 Biswas}, jest wystawiony na próbę przekształcenia normalnej misji pośrednika
16 w~spra~wach kupna i~sprzedaży nieruchomości, wraz z~bar~dziej lukratywnym
17 twierdzeniem dotyczącym jego życia i~nawyków, w~heral~dy~czną re~to~rykę
18 doświadczenia zarządzania kierowania losem: ',Dom Pana Bis~wasa''. Realistom,
19 doświadczanym tak jak Naipaul w~tak~tycznych b~łahostkach
20 codzienności, działanie to będzie wy~da~wać się nierozważne,
21 pretensjonalne.}\smallskip
22 \noindent Niewątpliwie.
23 \headline{\hfil\it Przykład I\newpageno}%\newpageno NIE JEST STANDARDOWĄ
24 \KOMENDĄ \TeX-a
25 \vfil\ejct

```

UWAGI

1, 5, 7, itd.: Wiązania (tyldy ~) zapobiegają podziałowi wiersza pomiędzy określonymi słowami. W wierszu 1 wiązanie jest wykorzystane w tym celu, żeby ostatnie słowo nie zostało przeniesione do nowej linii. Inny przykład zastosowania wiązań: niezręcznie jest umieszczać inicjały lub tytuł np. dr w jednej linii, a nazwisko w drugiej — wstawienie tzw. twardej spacji zapobiega tego typu sytuacjom. Komenda ta była omawiana także na str. 2.

2: Ujemna wartość parametrów komend `\leftskip` i `\rightskip` powoduje zmianę położenia krawędzi tekstu. Można także łączyć ujemne i dodatnie wartości w celu uzyskania eleganckiego efektu: jedna krawędź do wewnątrz, druga na zewnątrz.

12–13: Inną komendą służącą do kształtowania akapitu jest `\parshape`. Pierwsza liczba (powiedzmy n) określa liczbę wierszy do formatowania, dalej występują pary liczb (musi być n par). Pierwsza liczba w każdej z par określa wcięcie od lewego marginesu, a druga — długość wiersza. Jeśli jest mniej par niż wierszy w akapicie, ostatnie wiersze zostaną sforma-

towane według ustawień podanych w ostatniej z par. Jeśli jest zbyt wiele par, dodatkowe będą ignorowane. Komenda ta ma znaczenie w składzie czasopism i gazet: tekst można kształtować dowolnie, tworzyć nieregularne odstępy w celu dopasowania go do fotografii lub rysunków o nieregularnym kształcie. Na końcu każdego akapitu automatycznie przywracane jest domyślne ustawienie `\parshape=0`.

14: Jednym ze sposobów likwidacji wcięcia akapitu jest ustawienie parametru `\parindent` na 0. Należy pamiętać o właściwym grupowaniu tekstu (wykorzystując `{ }`) w celu uzyskania efektu nowego ustawienia.

16, 18: `\-` opcjonalne dzielenie (*discretionary hyphen*) sugeruje możliwe dzielenie wyrazów, w miejscach, których T_EX nie wybrał w tym celu. W tym przypadku sugestie wstawiono w celu łatwiejszego dopasowania dziwnego kształtu akapitu.

23: Ta komenda przywraca nagłówek. Patrz omówienie na str. 2 i 4.

Miał rację. A oto kilka początkowych wierszy z wielkiej powieści³:

Dziesięć tygodni przed śmiercią, Mohun Biswas, dziennikarz z Sikkim Street, St James, Port of Spain, został zwolniony z pracy. Chorował od jakiegoś czasu. Przed niespełna rokiem spędził ponad 9 tygodni w szpitalu Colonial, potem dochodził długo do zdrowia w domu. Kiedy lekarz zalecił mu całkowity wypoczynek, Trinidad Sentinel nie miało wyboru. Pan Biswas dostał 3-miesięczne wypowiedzenie, ale aż do śmierci był zaopatrywany każdego ranka w darmowy egzemplarz gazety.

I bardziej typowy rodzaj piśmiennictwa, jaki spotykamy dzisiaj (szczególnie w czasopiśmie naukowych) — początkowy fragment eseju napisanego przez krytyka literackiego, dotyczący wyżej cytowanej powieści⁴:

Ktoś,

czytając powieść V.S. Naipaula

Dom dla Pana Biswas, jest wystawiony

na próbę przekształcenia normalnej misji pośrednika w sprawach kupna i sprzedaży nieruchomości, wraz z bardziej lukratywnym twierdzeniem dotyczącym jego życia i nawyków, w heraldyczną retorykę doświadczenia zarządzania kierowania losem: „Dom Pana Biswasa”. Realistom, doświadczanym tak jak Naipaul w taktycznych błahostkach codzienności, działanie to będzie wydawać się nierozważne, pretensjonalne.

Niewątpliwie.

```

1 \centerline{\bf II. $\sqrt{2}$ --- LICZBA NIEWYMIERNA}
2 \centerline{\sl Eksponowanie równań; ułamki; potęgi; pierwiastki; przypisy.}
3 \headline{\hfil\it Przykład II\newpageno}\%newpageno --- KOMENDA WŁASNA
4 \bigskip
5 \noindent Liczby 1, 2, 3, \dots nazywamy {\sl liczbami naturalnymi}. Liczbę
6 nazywamy {\sl wymierną}, jeśli można ją zapisać w postaci ułamka  $m/n$  lub
7  $m/n$  (gdzie  $m$ ,  $n$  są liczbami naturalnymi), {\sl niewymierną} zaś ---
8 w przeciwnym wypadku. Pierwiastek stopnia  $n$  z liczby  $a$  oznaczamy symbolem
9  $\sqrt[n]{a}$ , gdzie  $n$  jest liczbą naturalną, i definiujemy jako dodatnią
10 liczbę  $b$ , taką że
11  $b^n = a$ , \quad {\rm tzn.}; \overbrace{b \times \dots \times
12 b}^n; {\rm razy}=a.
13 \bigbreak
14 \noindent{\it TWIERDZENIE 1}: Jeśli  $q > 1$ , to  $q^n > q$ , gdzie  $n$  jest dowolną
15 liczbą naturalną  $n \geq 2$ . \hfil\break
16 {\it DOWÓD}: Pomnożmy obie strony nierówności  $q > 1$  przez  $q$ . Mnożąc
17 otrzymaną nierówność  $q^2 > q$  ponownie przez  $q$ , dostajemy  $q^3 > q^2$ , a więc
18  $q^3 > q$ . Kontynuując w ten sam sposób, dochodzimy do wyniku
19  $q^n > q$ . \thinspace\clubsuit\footnote{\dag{Symbol  $\clubsuit$  oznacza tu
20 koniec dowodu. Zwykle symbolem końca dowodu jest kwadrat; w dodatku pod hasłem
21 {\sl kreska} znajduje się opis, jak należy go złożyć.}}
22 \bigbreak
23 Załóżmy teraz, że liczbę naturalną  $n$  można zapisać w postaci iloczynu  $n = m_1
24 \times m_2$ , gdzie  $m_1$  i  $m_2$  są liczbami naturalnymi większymi od 1. Wtedy
25  $n$  nazywamy {\sl liczbą złożoną}, a  $m_1$  i  $m_2$  --- {\sl czynnikami}  $n$ .
26 Jeśli liczby naturalnej  $p$  ( $p \neq 1$ ) nie da się zapisać w ten sposób,
27 mówimy, że jest to {\sl liczba pierwsza}. Przykładami liczb złożonych są
28 liczby 6 ( $= 2 \times 3$ ) i  $12$  ( $= 3 \times 4$ , lub  $= 2 \times 6$ ). Liczby pierwsze
29 to np. 2, 3, 7 i  $1013$ .
30 \bigbreak
31 \noindent{\it TWIERDZENIE 2}: Dowolną liczbę naturalną  $n$  większą od 1 można
32 zapisać w postaci iloczynu czynników pierwszych, tzn.  $n = p_1 \times p_2 \times
33 \dots \times p_k$ , gdzie  $p_i$  są liczbami pierwszymi. Ten rozkład na czynniki
34 jest jedyny. \vfil\ejct

```

UWAGI

KOMENTARZ

Na str. 26 zostanie omówione zastosowanie znaku $\$$ w uwagach.

1: \sqrt{x} składa \sqrt{x} .

3: `\headline` definiuje nagłówek. Patrz str. 2 i 4.

5: `\dots` składa ... (wielokropek matematyczny).

7, itd.: Wszystkie zmienne są ujęte w znaki $\$$, nawet jeśli występują w tekście. To zapewnia jednolitość kroju (znak $\$$ włącza i wyłącza kursywę matematyczną).

9: Komendy `\root` i `\of` pozwalają na skład dowolnych pierwiastków w trybie matematycznym, np. $\sqrt[n]{x}$ składa $\sqrt[n]{x}$. Nawiasy są konieczne tylko wtedy, gdy pod pierwiastkiem znajduje się więcej niż jeden znak.

11–12: Podwójny znak $\$$ (przed i za wyrażeniem) włącza i wyłącza eksponowany tryb matematyczny. Wyrażenie jest automatycznie umieszczane w osobnym wierszu, wyśrodkowane, z odstępami nad i pod wyrażeniem. Komenda `^` składa frakcję górną. `\rm` pozwala na skład tekstu antykwą w obrębie trybu matematycznego. Klamry `{}` określają, co ma wystąpić

w kroju `\rm`. `\;` określa odległość w trybie matematycznym, mały odstęp. \overbrace{A}^B składa nawias nad A, a B umieszcza nad nawiasem. `\quad` jest komendą odstępu poziomego. `\times` składa znak \times , a `\cdots` (wielokropek matematyczny podniesiony na wysokość znaku równości). **13, 22, itd.:** `\bigbreak` sugeruje T_EX-owi możliwość złamania strony. Jeśli to nie jest dobre miejsce na złamanie strony, T_EX w tym miejscu umieści odstęp pionowy równy `\bigskip`.

15: $x \geq y$ składa (w spolszczonym formacie; w oryginale jest to komenda \geq — przyp. tłum.).

19: `\thinspace` wstawia mały odstęp, takiej [] szerokości. `\clubsuit` składa symbol \clubsuit . `\footnote` składa przypis. Należy podać dwie informacje: symbol wykorzystany do oznaczenia przypisu i jego treść. Tutaj komenda symbolu `\dag` to znak \dagger . Tekst przypisu jest ujęty w nawiasy.

24, itd.: `_` składa frakcję dolną.

26: `\not` jest komendą w trybie matematycznym, neguje relację po nim następującą.

28: Zwróć uwagę, jak zostało złożone słowo „lub”.

II. $\sqrt{2}$ — LICZBA NIEWYMIERNA

Ekspozowanie równań; ułamki; potęgi; pierwiastki; przypisy.

Liczby 1, 2, 3, ... nazywamy *liczbami naturalnymi*. Liczbę nazywamy *wymierną*, jeśli można ją zapisać w postaci ułamka m/n lub $-m/n$ (gdzie m, n są liczbami naturalnymi), *niewymierną* zaś — w przeciwnym wypadku. Pierwiastek stopnia n z liczby a oznaczamy symbolem $\sqrt[n]{a}$, gdzie n jest liczbą naturalną, i definiujemy jako dodatnią liczbę b , taką że

$$b^n = a, \quad \text{tzn. } \overbrace{b \times \cdots \times b}^{n \text{ razy}} = a.$$

TWIERDZENIE 1: Jeśli $q > 1$, to $q^n > q$, gdzie n jest dowolną liczbą naturalną ≥ 2 .

DOWÓD: Pomnożmy obie strony nierówności $q > 1$ przez q . Mnożąc otrzymaną nierówność $q^2 > q$ ponownie przez q , dostajemy $q^3 > q^2$, a więc $q^3 > q$. Kontynuując w ten sam sposób, dochodzimy do wyniku $q^n > q$. ♣†

Założmy teraz, że liczbę naturalną n można zapisać w postaci iloczynu $n = m_1 \times m_2$, gdzie m_1 i m_2 są liczbami naturalnymi większymi od 1. Wtedy n nazywamy *liczbą złożoną*, a m_1 i m_2 — *czynnikami* n . Jeśli liczby naturalnej p ($p \neq 1$) nie da się zapisać w ten sposób, mówimy, że jest to *liczba pierwsza*. Przykładami liczb złożonych są liczby 6 ($= 2 \times 3$) i 12 ($= 3 \times 4$ lub $= 2 \times 6$). Liczby pierwsze to np. 2, 3, 7 i 1013.

TWIERDZENIE 2: Dowolną liczbę naturalną n większą od 1 można zapisać w postaci iloczynu czynników pierwszych, tzn. $n = p_1 \times p_2 \times \cdots \times p_k$, gdzie p_i są liczbami pierwszymi. Ten rozkład na czynniki jest jedyny.

† Symbol ♣ oznacza tu koniec dowodu. Zwykle symbolem końca dowodu jest kwadrat; w dodatku pod hasłem *kreska* znajduje się opis, jak należy go złożyć.


```

1 \noindent{\it DOWÓD\}/: Gdy  $n$  jest liczbą pierwszą, problem jest rozwiązany.
2 Jeśli jest liczbą złożoną, to  $n=m_1 \cdot m_2$ ,  $m_1 > 1$ ,  $m_2 > 1$ ,  $n > m_1$  i
3  $n > m_2$ . Proces rozkładu na czynniki powtarzamy dla  $m_1$  i  $m_2$ . Ponieważ
4  $n$  jest skończone, proces taki zakończy się podziałem  $n$  na czynniki pierwsze.
5 \bigbreak
6 Przepuścimy\footnote{\^{\star}}{Ta część dowodu została zaczerpnięta z książki:
7 R. Courant i H. Robbins, {\sl What is Mathematics?}, Oxford University Press
8 (1941).}, że  $n$  jest najmniejszą liczbą naturalną, posiadającą dwa różne
9 rozkłady na czynniki pierwsze, tzn.  $n=p_1 \cdot \dots \cdot p_k = q_1 \cdot \dots \cdot q_l$ 
10  $\cdot \dots \cdot q_l \cdot \dots \cdot q_l \cdot \dots \cdot q_l \cdot \dots \cdot q_l$  gdzie  $p_i$  i  $q_i$  są liczbami pierwszymi
11 ( $p_i \neq q_i$ ). Niech  $p_1$  i  $q_1$  będą najmniejszymi czynnikami rozkładu.
12 Wtedy  $p_1 \neq q_1$ . Załóżmy, że  $p_1 < q_1$ . Niech  $n' = n - (p_1 \cdot \dots \cdot q_l)$ 
13  $\cdot \dots \cdot q_l \cdot \dots \cdot q_l \cdot \dots \cdot q_l \cdot \dots \cdot q_l$ .
14 Wtedy mamy  $n' = \begin{cases} p_1(p_2 \cdot \dots \cdot p_k - q_2 \cdot \dots \cdot q_l) \\ q_1(p_1 - q_1)(q_2 \cdot \dots \cdot q_l) \end{cases}$ 
15 & pierwszy rozkład na czynniki, \cr & \cr
16  $(q_1 - p_1)(q_2 \cdot \dots \cdot q_l)$  & drugi rozkład na czynniki. \cr} \eqno(3)
17 \smallskip\noindent
18 Niech  $m = q_2 \cdot \dots \cdot q_l$ . Ponieważ  $m < n$ ,  $m$  i  $n'$  mają
19 jednoznaczny rozkład na czynniki pierwsze. Z warunku (3) wynika, że  $p_1$  jest
20 czynnikiem  $n' = (q_1 - p_1)m$ , ale nie może być czynnikiem  $m$ , ponieważ jest
21 mniejsze od każdego  $q_j$ . Jeśli  $p_1$  byłoby czynnikiem  $(q_1 - p_1)$ , to
22  $q_1 - p_1 = gp_1$ . Stąd mielibyśmy  $q_1 = (g+1)p_1$ , co przeczy założeniu, że  $q_1$ 
23 jest liczbą pierwszą. \thinspace\clubsuit$
24 \bigbreak
25 \noindent{\it WNIOSEK\}/: Jeśli liczba pierwsza  $p$  jest czynnikiem iloczynu
26  $a \cdot b$ , to jest równocześnie czynnikiem  $a$  lub  $b$ .
27 \bigbreak
28 \noindent{\it TWIERDZENIE 3\}/: Niech  $m$  i  $n$  będą liczbami naturalnymi.
29 Jeśli  $\sqrt[m]{n}$  jest liczbą naturalną, gdzie  $p$  i  $q$  są liczbami naturalnymi,
30 takimi że  $q$  nie jest czynnikiem  $p$ , to  $q=1$ . \hfil\break
31 \vfil\ejct

```

UWAGI

KOMENTARZ

Większość z komend znajdujących się tu była omówiona na str. 24. Komendy trybu matematycznego prezentowane w uwagach są umieszczone pomiędzy znakami $\$$. Znaki te w tym miejscu są jedynie przypomnieniem: kiedy pracujesz w trybie matematycznym, nie należy umieszczać wielu komend w ten sposób, jedynie w całych wzorach, jak w pudełku powyżej.

6: $\$ \star$ składa znak \star . $\$ \text{footnote}$ tu trochę różni się od przypisu z poprzedniej strony. Znak \star , oznaczający przypis, jest tu użyty we frakcji górnej, za pomocą komendy $\^$. Nawiasy $\{ i \}$ określają, w którym miejscu kończy się definicja symbolu i gdzie jest początek przypisu. Krój pisma w przypisach można zmieniać tak jak w tekście głównym.

7: Znaki tyldy \sim to wiązania. Tworzą one odstęp między wyrazami i zapobiegają podziałowi wiersza w danym miejscu. Patrz str. 2 i 22.

10: $\$ \text{eqno}$ wykorzystane jest do numerowania równań eksponowanych. Aby numer równania znajdował się po lewej stronie, należy zastosować komendę $\$ \text{leqno}$ (patrz przykład VII).

9, 12, itd.: Podwójny znak $\$$ określa początek i koniec eksponowanych wyrażeń matematycznych. Takie eksponowane wyrażenia są automatycznie wyśrodkowane w nowym wierszu oraz dołożony jest odstęp nad i pod wyrażeniem. Patrz także w dodatku pod hasłami np. *matematyka: wyrażenia eksponowane* i *matematyka: odstępy*.

14–16: $\$ \text{cases}$ pozwala wyrównywać wielowierszowe wyrażenia eksponowane. Każdy wiersz eksponowanego wyrażenia składa się z dwóch części: pierwsza to wyrażenie w trybie matematycznym, druga — zwykły tekst. Części są oddzielone znakiem $\&$, każdy z wierszy zakończony jest komendą $\$ \text{cr}$ (*carriage return*). Tekstu komentarza nie musimy wpisywać, ale znak $\&$ musi pojawić się obowiązkowo. Na przykład eksponowane równanie (3) jest w rzeczywistości trzyliniowym wyrażeniem eksponowanym, z pustym wierszem środkowym (zrobiono to w celu uzyskania odstępu pionowego). Struktura tu przedstawiona jest charakterystyczna dla justowanych w $\text{T}_{\text{E}}\text{X}$ -u wyrażeń eksponowanych (jak macierze, tabele); znak $\&$ gra rolę funkcji „tabularyzującej”, przeskoku od wpisu do wpisu, a $\$ \text{cr}$ kończy wiersz.

DOWÓD: Gdy n jest liczbą pierwszą, problem jest rozwiązany. Jeśli jest liczbą złożoną, to $n = m_1 \times m_2$, $m_1 > 1$, $m_2 > 1$, $n > m_1$ i $n > m_2$. Proces rozkładu na czynniki powtarzamy dla m_1 i m_2 . Ponieważ n jest skończone, proces taki zakończy się podziałem n na czynniki pierwsze.

Przypuśćmy*, że n jest najmniejszą liczbą naturalną, posiadającą dwa różne rozkłady na czynniki pierwsze, tzn.

$$n = p_1 \times \cdots \times p_k = q_1 \times \cdots \times q_l \quad (1)$$

gdzie p_i i q_i są liczbami pierwszymi ($p_i \neq q_i$). Niech p_1 i q_1 będą najmniejszymi czynnikami rozkładu. Wtedy $p_1 \neq q_1$. Załóżmy, że $p_1 < q_1$. Niech

$$n' = n - (p_1 \times q_2 \times \cdots \times q_l). \quad (2)$$

Wtedy mamy

$$n' = \begin{cases} p_1(p_2 \times \cdots \times p_k - q_2 \times \cdots \times q_l) & \text{pierwszy rozkład na czynniki,} \\ (q_1 - p_1)(q_2 \times \cdots \times q_l) & \text{drugi rozkład na czynniki.} \end{cases} \quad (3)$$

Niech $m = q_2 \times \cdots \times q_l$. Ponieważ $m < n' < n$, m i n' mają jednoznaczny rozkład na czynniki pierwsze. Z warunku (3) wynika, że p_1 jest czynnikiem $n' = (q_1 - p_1)m$, ale nie może być czynnikiem m , ponieważ jest mniejsze od każdego q_j . Jeśli p_1 byłoby czynnikiem $(q_1 - p_1)$, to $q_1 - p_1 = gp_1$. Stąd mielibyśmy $q_1 = (g+1)p_1$, co przeczy założeniu, że q_1 jest liczbą pierwszą. ♣

WNIOSEK: Jeśli liczba pierwsza p jest czynnikiem iloczynu $a \times b$, to jest równocześnie czynnikiem a lub b .

TWIERDZENIE 3: Niech m i n będą liczbami naturalnymi. Jeśli $\sqrt[n]{m} = p/q$, gdzie p i q są liczbami naturalnymi, takimi że q nie jest czynnikiem p , to $q = 1$.

* Ta część dowodu została zaczerpnięta z książki: R. Courant i H. Robbins, *What is Mathematics?*, Oxford University Press (1941).

```

1 \noindent{\it DOWÓD\}/: Załóżmy, że  $q \neq 1$ . Wtedy mamy
2 
$$\sqrt[n]{m} = \frac{p_1 \cdots p_k}{q_1 \cdots q_l}, \text{ \eqno(4)}$$

3
4 gdzie  $p_1, \dots, p_k$  i  $q_1, \dots, q_l$  są czynnikami pierwszymi,
5 odpowiednio,  $p$  i  $q$ . Jeśli jakikolwiek czynnik w liczniku jest równy
6 czynnikowi z mianownika, obydwie można skreślić. Przypuśćmy, że taka operacja
7 została wykonana. Wtedy podnosząc równanie (4) do  $n$ -tej potęgi, otrzymujemy
8 
$$m = \frac{(p_1 \cdots p_k)^n}{(q_1 \cdots q_l)^n},$$

9 stąd mamy
10 
$$(q_1 \cdots q_l)^n m = (p_1 \cdots p_k)^n.$$

11 Ponieważ  $q_1$  jest czynnikiem lewej strony równania, musi być czynnikiem
12 jednego z  $p_i$  z prawej strony równania. Przeczy to jednak założeniu. Tak więc
13 musi być  $q=1$ . \thinspace\clubsuit$
14 \bigbreak
15 \noindent{\it WNIOSZEK\}/:  $\sqrt{2}$  jest liczbą niewymierną. \hfil\break
16 {\it DOWÓD\}/: Załóżmy, że  $\sqrt{2} = p/q$ , gdzie  $p$  i  $q$  są liczbami
17 naturalnymi. Z twierdzenia 3 otrzymujemy  $q=1$ . Zgodnie z twierdzeniem 1,
18  $p^2 = 2q^2$ . Ale  $p^2 = 2q^2$ . Zatem  $p$  byłoby liczbą naturalną mniejszą niż 2, czyli
19  $p=1$ , wtedy mielibyśmy  $\sqrt{2} = 1$ , co nie jest prawdą. \thinspace\clubsuit$
20 \vfil\ejct

```

UWAGI

2: Komenda `\over` składa ułamek. Struktura tej komendy jest prosta: $\text{\$licznik\over mianownik\$}$, gdzie licznik i mianownik mogą być dłuższymi wyrażeniami. Wszystkie pozostałe użyte tu komendy były wcześniej wyjaśnione.

2, 8, itd.: Patrz dyskusja na str. 26, dotycząca pracy w trybie matematycznym.

O POTĘGACH

Komenda `^` składa znak po niej następujący jako frakcję górną; dalsze znaki są traktowane jako część wyrażenia głównego (wiersz 10). Aby złożyć dłuższe wyra-

żenie jako frakcję górną, należy wpisać `^{\text{\$długie wyrażenie\$}}`. Te same reguły obowiązują w składzie frakcji dolnej.

Zapis `x^{y^z}` jest właściwym sposobem na skład potęgi w potęgę; `{x^y}^z` daje dziwny efekt składu (rozmiar `z` jest niepoprawny i wyrażenie wygląda trochę tak, jak gdyby złożone było w postaci `x^{yz}`). Zazwyczaj trzeba trochę poeksperymentować przed wybraniem formy czytelnej, a jednocześnie najlepiej wyrażającej to, o co nam chodzi. (Porównaj np. efekty działania komend: $(x^y)^z$ i $\{(x^y)\}^z$.)

DOWÓD: Załóżmy, że $q \neq 1$. Wtedy mamy

$$\sqrt[n]{m} = \frac{p}{q} = \frac{p_1 \times \cdots \times p_k}{q_1 \times \cdots \times q_l}, \quad (4)$$

gdzie p_1, \dots, p_k i q_1, \dots, q_l są czynnikami pierwszymi, odpowiednio, p i q . Jeśli jakikolwiek czynnik w liczniku jest równy czynnikowi z mianownika, obydwa można skreślić. Przypuśćmy, że taka operacja została wykonana. Wtedy podnosząc równanie (4) do n -tej potęgi, otrzymujemy

$$m = \frac{(p_1 \times \cdots \times p_k)^n}{(q_1 \times \cdots \times q_l)^n},$$

stąd mamy

$$(q_1 \times \cdots \times q_l)^n m = (p_1 \times \cdots \times p_k)^n.$$

Ponieważ q_1 jest czynnikiem lewej strony równania, musi być czynnikiem jednego z p_i z prawej strony równania. Przeczy to jednak założeniu. Tak więc musi być $q = 1$. ♣

WNIOSEK: $\sqrt{2}$ jest liczbą niewymierną.

DOWÓD: Załóżmy, że $\sqrt{2} = p/q$, gdzie p i q są liczbami naturalnymi. Z twierdzenia 3 otrzymujemy $q = 1$. Zgodnie z twierdzeniem 1, $p < p^2$. Ale $p^2 = 2$. Zatem p byłoby liczbą naturalną mniejszą niż 2, czyli $p = 1$, wtedy mielibyśmy $\sqrt{2} = p = 1$, co nie jest prawdą. ♣

```

1 \centerline{\bf III. POZIOMY CAUCHY'EGO}
2 \centerline{\sl Symbole teorii zbiorów; puste odstępy na schematy.}
3 \headline{\hfil\t Przykład III\newpageno}\% \newpageno --- KOMENDA WŁASNA
4 \bigskip \noindent
5 Niech  $\mathcal{M}$  będzie czasowo=orientowaną czasoprzestrzenią (tzn.
6 czasoprzestrzenią o globalnym wyborze kierunku czasu przyszłego). {\sl Krzywą
7 typu czasowego (zerową)\} w  $\mathcal{M}$  definiujemy jako krzywą  $C^1$ , do której
8 styczna jest wszędzie zerowa (typu czasowego). Krzywa typu czasowego
9 jest zamknięta, jeśli nie posiada punktów końcowych. Wtedy, dla każdego
10  $p \in \mathcal{M}$ , {\sl przyszłość (przeszłość) chronologiczną\}  $p$  określamy
11 następująco:
12 \smallskip
13 \centerline{ $I^{\pm}(p) = \{q \in \mathcal{M} \mid \exists s \text{ skierowana w przyszłość}$ 
14  $(\text{przeszłość}) \text{ krzywa typu czasowego od } p \text{ do } q\}$ .}
15 \smallskip \noindent
16 Definicję tę można rozszerzyć na dowolną  $A \subset \mathcal{M}$ :
17  $I^{\pm}(A) = \bigcup_{p \in A} I^{\pm}(p)$ .
18 \indent Niech teraz  $S \subset \mathcal{M}$  będzie achronalną hiperpowierzchnią
19 typu przestrzennego (tzn. żadnych jej dwóch punktów nie można połączyć krzywą
20 typu czasowego) bez krawędzi. Zdefiniujmy następująco {\sl przyszłą (przeszłą)
21 dziedzinę zależności\} dla  $S$ : \medskip
22 {\narrower\itemitem{ $D^{\pm}(S) = \{p \in \mathcal{M} \mid \forall s \text{ pełnej}$ 
23  $\text{krzywej typu czasowego lub krzywej zerowej } \rho, \text{ przechodzącej przez } p,$ 
24  $\rho \cap S \neq \emptyset, z \text{ } S \text{ leżącym po stronie przeszłości}$ 
25  $(\text{przyszłości}) p \text{ na } \rho\}$ . \medskip}
26 \noindent W przypadku hiperpowierzchni  $S$  typu przestrzennego, może się
27 zdarzyć, że  $I^{\pm}(S) - D^{\pm}(S) \neq \emptyset$ ; tzn. mogą
28 istnieć punkty leżące w przyszłości (przeszłości) względem  $S$ , które nie
29 leżą w  $D^{\pm}(S)$ . Wówczas  $D^{\pm}(S)$  ma niepusty brzeg
30 {\sl przyszłości\} ({\sl przeszłości\}), nazywany {\sl poziomem Cauchy'ego},
31 zdefiniowany jako \smallskip
32 \centerline{ $H^{\pm}(S) =$ 
33  $\overline{D^{\pm}(S)} - I^{\pm}(D^{\pm}(S))$ .} \smallskip
34 \noindent Wszystkie te zbiory przedstawione są na rysunku~1.
35 \pageinsert
36 Opis tej wstawki znajduje się na stronie następnej.
37 \endinsert
38 Należy je rozumieć następująco: powierzchnię  $S$  można traktować jako
39 powierzchnię wartości początkowych, na której można zadawać stan początkowy
40 jakiegoś systemu (który sam może być geometrią czasoprzestrzeni).
41  $D^+(S)$  przedstawia obszar, w którym przyszły rozwój systemu można
42 wyliczyć na podstawie danych początkowych. Istnieją analogiczne, ale odwrócone
43 w czasie, interpretacje  $D^-(S)$ .

```

UWAGI

KOMENTARZ

Komendy trybu matematycznego omówione w uwagach są umieszczone pomiędzy znakami $\$$, które w tym miejscu stanowią jedynie przypomnienie; podczas pracy w trybie matematycznym nie należy umieszczać wielu komend w ten sposób, z wyjątkiem całych wzorów, jak w ramce powyżej.

5: $\backslash\mathcal{}$ odnosi się do kaligrafowanego kroju pisma w trybie matematycznym (styl dostępny tylko w przypadku dużych liter). Należy wyraźnie grupować materiał do wykaligrafowania (tak jak to zrobiono w kilku wyrażeniach na tej stronie), w przeciwnym razie można uzyskać dziwne efekty składu; patrz komenda $\backslash\mathcal{}$ w dodatku.

10: $\backslash\in$ składa symbol \in .

13–14: $\backslash\pm$ składa \pm , $\backslash\{$ składa $\{$, $\backslash\mid$ składa $|$, $\backslash\exists$ składa \exists , a $\backslash\}$ składa $\}$.

16: $\backslash\subset$ składa \subset .

17: $\backslash\bigcup$ składa duży symbol sumy. Komenda frakcji dolnej $_$ jest wykorzystana tu do umieszczenia pewnego wyrażenia pod symbolem. W przykładzie VII przedstawiono sposób przemieszczania frakcji górnej lub dolnej z boku symbolu do góry i odwrotnie.

22–25: Komendy $\backslash\text{narrower}$ i $\backslash\text{itemitem}$ służą tu do wyeksponowania definicji. Celem składu było lewostronne wyrównanie tekstu następującego po znaku $=$, co uzyskano za pomocą komendy $\backslash\text{itemitem}$. W taki sam sposób działa $\backslash\text{item}$, ale daje o połowę mniejsze wcięcie niż $\backslash\text{itemitem}$.

22–24: $\backslash\forall$ składa \forall , $\backslash\rho$ składa ρ , $\backslash\cap$ składa \cap , a $\backslash\emptyset$ składa \emptyset .

33: $\backslash\overline{}$ składa kreskę nad materiałem zgrupowanym w nawiasach; $\backslash\mp$ składa znak \mp .

35–37: $\backslash\text{pageinsert}$ umieszcza daną wstawkę na następnej stronie. Komenda ta jest przydatna przy włączaniu rysunków i diagramów. $\backslash\text{endinsert}$ kończy umieszczenie.

III. POZIOMY CAUCHY'EGO

Symbole teorii zbiorów; puste odstępy na schematy.

Niech \mathcal{M} będzie czasowo-orientowaną czasoprzestrzenią (tzn. czasoprzestrzenią o globalnym wyborze kierunku czasu przyszłego). Krzywą typu czasowego (zerową) w \mathcal{M} definiujemy jako krzywą C^1 , do której styczna jest wszędzie zerowa (typu czasowego). Krzywa typu czasowego jest zamknięta, jeśli nie posiada punktów końcowych. Wtedy, dla każdego $p \in \mathcal{M}$, przyszłość (przeszłość) chronologiczną p określamy następująco:

$$I^\pm(p) = \{q \in \mathcal{M} \mid \exists \text{ skierowana w przyszłość (przeszłość) krzywa typu czasowego od } p \text{ do } q\}.$$

Definicję tę można rozszerzyć na dowolną $\mathcal{A} \subset \mathcal{M}$:

$$I^\pm(\mathcal{A}) = \bigcup_{p \in \mathcal{A}} I^\pm(p).$$

Niech teraz $\mathcal{S} \subset \mathcal{M}$ będzie achronalną hiperpowierzchnią typu przestrzennego (tzn. żadnych jej dwóch punktów nie można połączyć krzywą typu czasowego) bez krawędzi. Zdefiniujemy następująco przyszłą (przeszłą) dziedzinę zależności dla \mathcal{S} :

$$D^\pm(\mathcal{S}) = \{p \in \mathcal{M} \mid \forall \text{ pełnej krzywej typu czasowego lub krzywej zerowej } \rho, \text{ przechodzącej przez } p, \rho \cap \mathcal{S} \neq \emptyset, \text{ z } \mathcal{S} \text{ leżącym po stronie przeszłości (przyszłości) } p \text{ na } \rho\}.$$

W przypadku hiperpowierzchni \mathcal{S} typu przestrzennego, może się zdarzyć, że $I^\pm(\mathcal{S}) - D^\pm(\mathcal{S}) \neq \emptyset$; tzn. mogą istnieć punkty leżące w przyszłości (przeszłości) względem \mathcal{S} , które nie leżą w $D^\pm(\mathcal{S})$. Wówczas $D^\pm(\mathcal{S})$ ma niepusty brzeg przyszłości (przeszłości), nazywany poziomem Cauchy'ego, zdefiniowany jako

$$H^\pm(\mathcal{S}) = \overline{D^\pm(\mathcal{S})} - I^\mp(D^\pm(\mathcal{S})).$$

Wszystkie te zbiory przedstawione są na rysunku 1.

Należy je rozumieć następująco: powierzchnię \mathcal{S} można traktować jako powierzchnię wartości początkowych, na której można zadawać stan początkowy jakiegoś systemu (który sam może być geometrią czasoprzestrzeni).

Oto tekst dotyczący komendy `\pageinsert` z poprzedniej strony. Był on wprowadzony jako część głównego dokumentu, z komendą `\endinsert`, oznaczającą koniec wstawki. \TeX reaguje na te komendy przez umieszczenie materiału zawartego między nimi na następnej stronie. Może to być pusta strona, uzyskana w następujący sposób:

```
\pageinsert
\vfil
\endinsert
```

lub dowolna wstawka, jak w tym miejscu.

Pomiędzy nagłówkami rysunków pozostawiono puste miejsce na grafikę, `\vfil`, która została wykonana w innym programie (w tym przypadku w programie MetaPost — przyp. tłum.). Możliwy jest elektroniczny import plików graficznych z innych źródeł

i umieszczanie ich w tekście np. za pomocą komendy `\special` (komenda ta nie jest interpretowana przez \TeX -a, lecz umieszczana w pliku *.dvi wraz z argumentem, który jest znakiem sterującym czytelnym dla urządzenia drukującego; szczegóły można znaleźć w literaturze [1] — przyp. tłum.).

9: W tym miejscu pokazano, jak składać pierwiastek `\sqrt` obejmujący dłuższe wyrażenie.

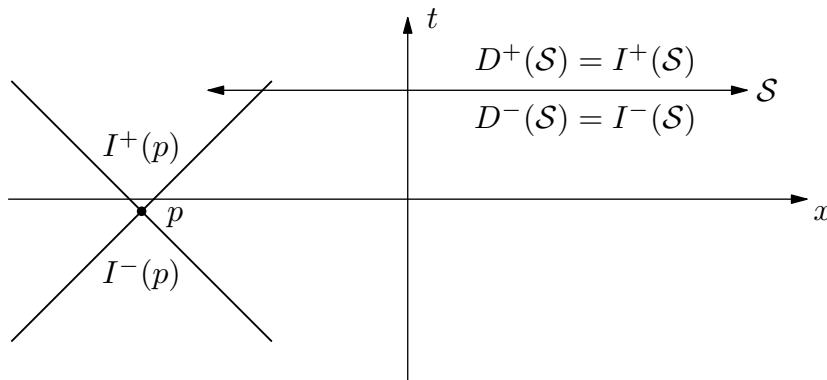
14: `\psi` składa ψ , a `\infty` składa ∞ . Komenda `\,` tworzy odstęp, małą spację pomiędzy $d\psi$ i dt .

15: `\xleq` składa \leq .

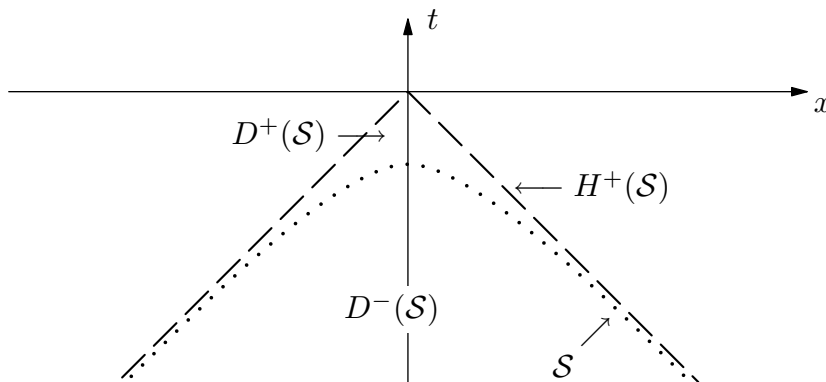
KOMENTARZ

Znaki `$` określają, które z komend pracują wyłącznie w trybie matematycznym.

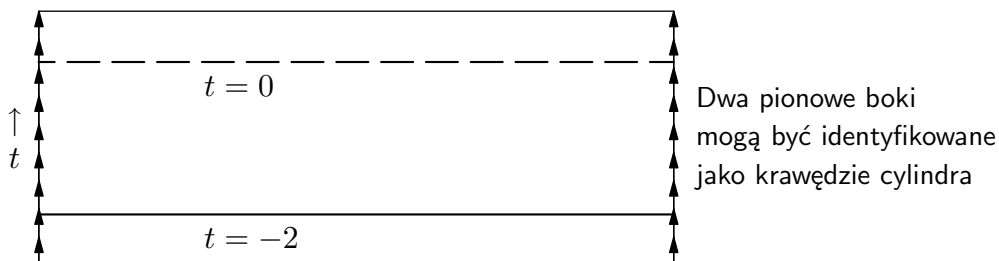
Rysunek 1



- 1a. Dwuwymiarowa przestrzeń Minkowskiego; $ds^2 = -dt^2 + dx^2$. \mathcal{S} jest powierzchnią $t = 1$. $D^\pm(\mathcal{S}) = I^\pm(\mathcal{S})$. Pokazane są także zbiory $I^\pm(p)$ dla pewnego $p \in \mathcal{M}$.



- 1b. Kolejny przykład dwuwymiarowej przestrzeni Minkowskiego; \mathcal{S} jest zadana przez $t = -\sqrt{1+x^2}$. $D^-(\mathcal{S}) = I^-(\mathcal{S})$, ale $D^+(\mathcal{S}) \neq I^+(\mathcal{S})$. Więc $H^+(\mathcal{S}) \neq \emptyset$.



- 1c. Dwuwymiarowa czasoprzestrzeń z cylindrem $R \times S^1$ jako rozmaitość; $ds^2 = -d\psi dt - t(d\psi)^2$, $-\infty < t < \infty$, $0 \leq \psi \leq 2\pi$. Tutaj $t = -2$ jest hiperpowierzchnią typu przestrzennego, a $t = 0$ jest przyszłym horyzontem Cauchy'ego. Ten horyzont Cauchy'ego jest zbiorem zwartym.


```

1 Poziomy Cauchy'ego przedstawiają krańce naszych możliwości przewidywania
2 przyszłości, nawet jeśli mamy kompletną wiedzę na temat teraźniejszości.
3 Istnienie czasoprzestrzeni nie posiadających żadnych powierzchni  $\int_{\mathcal{S}} d^3x$ ,
4 takich że  $D^{\mu}(\int_{\mathcal{S}} d^3x) = I^{\mu}(\mathcal{S})$  (czyli posiadających nietrywialne
5 poziomy Cauchy'ego), jako pierwszy prawdopodobnie odkrył i opisał Roger
6 Penrose5. Właściwości poziomów Cauchy'ego szeroko przebadał Stephen Hawking.
7 Wyniki jego badań przedstawione są w poz. 6. Przypuszcza się, że
8 czasoprzestrzenie, które zawierają takie poziomy Cauchy'ego, są bardzo
9 szczególne i nierealistyczne fizycznie. Uwidaczniają to badania zwartych
10 poziomów Cauchy'ego (takich jak na rysunku 1c)7,8.
11
12 \line{\hfil}
13 \smallbreak
14 \hrule height1pt
15 \smallskip
16 \hrule height1pt
17 \line{\hfil}
18 \bigskip
19 \centerline{\bf IV. HISZPAŃSKIE INTERLUDIUM}
20 \centerline{\sl Niezwykła interpunkcja; akcenty; komendy nakładania.}
21 \headline{\hfil\it Przykład IV\newpageno}\% \newpageno --- KOMENDA WŁASNA
22 \bigskip
23 \noindent Kilka lat temu Donald Knuth napisał zachwycającą książkę na temat
24 nowego systemu liczb (nazwanego przez niego systemem „liczb
25 nadrzeczywistych”), który został wynaleziony przez Johna Conwaya. Książka ma
26 formę powieści9. Kobieta i mężczyzna odpoczywają w opuszczonym miejscu, aby
27 „odnaleźć siebie”. Tam przypadkowo odkrywają zasady konstruowania liczb
28 Conwaya. Są zaintrygowani i zaczynają badać właściwości tych nowych liczb.
29 Podczas pracy ich zainteresowanie wzrasta, stają się coraz bardziej podnieceni.
30 I wreszcie podniecenie to przestaje być wyłącznie intelektualne.
31
32 Książka Knutha jest prawdopodobnie pierwszym opisem zaawansowanej matematyki,
33 zawierającym elementy seksu. Stosowny fragment (bardzo interesujący)
34 został zacytowany poniżej, w wersji hiszpańskiej\footnote{$^{\ast}$}{Tłumaczenie
35 --- Luca Bombelli.}:
36

```

UWAGI

7, 10: Po raz kolejny w kilku miejscach wykorzystano wiązania. Wyjaśnienie tej komendy znajduje się na str. 2, 22 i 26.

12: `\line{\hfil}` składa wiersz wypełniony justunkiem.

13: `\smallbreak` sugeruje TeX-owi złamanie strony. Jeśli sugestia taka nie jest w tym momencie uwzględniona, program wstawia pionowy odstęp równy `\smallskip`.

14, 16: Komendy `\hrule` składają kreski poziome o wysokości 1 punktu i szerokości kolumny tekstu.

15: `\smallskip` składa odstęp między kreskami.

17–18: Kolejny pusty wiersz, następujący przed odstępem pionowym `\bigskip`.

21: Komenda `\headline` pojawia się tu w środku strony, w celu zmiany jej nagłówka (TeX ustawia nagłówki strony zgodnie z ostatnią komendą `\headline` — dyskusja na str. 4). `\newpageno` to specjalna komenda, potrzebna przy formatowaniu tej książki. Patrz str. 2.

34: Jeszcze jeden przypis, nic nowego.

36: W tym miejscu jest łamana strona, pomiędzy akapitami nie pojawia się jednak żaden dodatkowy odstęp, ani na dole tej strony, ani u góry następnej.

$D^+(\mathcal{S})$ przedstawia obszar, w którym przyszły rozwój systemu można wyliczyć na podstawie danych początkowych. Istnieją analogiczne, ale odwrócone w czasie, interpretacje $D^-(\mathcal{S})$. Poziomy Cauchy'ego przedstawiają krańce naszych możliwości przewidywania przyszłości, nawet jeśli mamy kompletną wiedzę na temat teraźniejszości. Istnienie czasoprzestrzeni nie posiadających żadnych powierzchni \mathcal{S} , takich że $D^\pm(\mathcal{S}) = I^\pm(\mathcal{S})$ (czyli posiadających nietrywialne poziomy Cauchy'ego), jako pierwszy prawdopodobnie odkrył i opisał Roger Penrose⁵. Właściwości poziomów Cauchy'ego szeroko przebadał Stephen Hawking. Wyniki jego badań przedstawione są w poz. 6. Przypuszcza się, że czasoprzestrzenie, które zawierają takie poziomy Cauchy'ego, są bardzo szczególne i nierealistyczne fizycznie. Uwidaczniają to badania zwartych poziomów Cauchy'ego (takich jak na rysunku 1c)^{7,8}.

IV. HISZPAŃSKIE INTERLUDIUM

Niezwykła interpunkcja; akcenty; komendy nakładania.

Kilka lat temu Donald Knuth napisał zachwycającą książkę na temat nowego systemu liczb (nazwanego przez niego systemem „liczb nadrzeczywistych”), który został wynaleziony przez Johna Conwaya. Książka ma formę powieści⁹. Kobieta i mężczyzna odpoczywają w opuszczonym miejscu, aby „odnaleźć siebie”. Tam przypadkowo odkrywają zasady konstruowania liczb Conwaya. Są zaintrygowani i zaczynają badać właściwości tych nowych liczb. Podczas pracy ich zainteresowanie wzrasta, stają się coraz bardziej podnieceni. I wreszcie podniecenie to przestaje być wyłącznie intelektualne.

Książka Knutha jest prawdopodobnie pierwszym opisem zaawansowanej matematyki, zawierającym elementy seksu. Stosowny fragment (bardzo interesujący) został zacytowany poniżej, w wersji hiszpańskiej*:

B: Alicia, estoy descubriendo un nuevo aspecto de tu personalidad hoy. Realmente derribas el mito de que las mujeres no valen para las matemáticas.

* Tłumaczenie — Luca Bombelli.

```

1 \item{B:}{Alicia, estoy descubriendo un nuevo aspecto de tu personalidad
2 hoy. Realmente derribas el mito de que las mujeres no valen para las
3 mate\m\'aticas.}
4 \item{A:}{!‘Pues muchas gracias, amable caballero!}
5 \item{B:}{S\’e que suena como una locura, pero trabajar contigo en estas
6 cosas creativas me hace sentir como un potro salvaje. Uno pensar\’\i a que
7 tanto trabajar de seso apagar\’\i a cualquier deseo f\’\i sico, pero en
8 realidad hace mucho tiempo que no me sentia as\’\i.}
9 \item{A:}{Si quieres que te diga la verdad, yo tambi\’en.}
10 \item{B:}{Mira la puesta de sol, es igual que el poster que compramos
11 aquella vez. Y mira el agua.}
12 \item{A:}{(Corriendo) Vamos.}\vskip1pt
13 \line{\llap{\$heartsuit\heartsuit}\hfil\dots\dots\dots\hfil
14 \rlap{\$heartsuit\heartsuit}}
15 \item{B:}{!‘Jol\’\i n! Nunca hab\’\i a dormido tan bien.}
16 \item{A:}{Lo mismo digo yo. Es estupendo despertat y estar realmente
17 despierta, no solamente ‘‘despierta de caf\’e’’}
18 \item{B:}{?‘Donde hab\’\i amos quedado ayer, antes de que perdi\’eramos la
19 cabeza y nos olvid\’aramos por completo de las matem\’aticas?}
20
21 \line{}\smallbreak
22 \hrule height1pt\smallskip\hrule height1pt
23 \line{}\bigskip
24 \centerline{\bf V. SPULAPKOWANE POWIERZCHNIE}
25 \centerline{\bf W PRZESTRZENI G"ODLA}
26 \centerline{\sl Macierze; justowanie r3wnań; indeksy; pochodne
27 cząstkowe; przypisy.}
28 \headline{\hfil\it Przykład V\newpageno}\%newpageno --- komenda własna.
29 \bigskip
30 \noindent W 1949 roku Kurt G"odel przedstawił kosmologiczne rozwiązanie
31 równania Einsteina z pewnymi bardzo dziwnymi własnościami\^{10}\$. Wydaje się, że
32 G"odel odkrył to rozwiązanie w trakcie badania idealistycznej koncepcji czasu,
33 tj. problemu, czy ,,rzeczywistość składa się z nieskończonej liczby warstw
34 teraźniejszości’’. W jego modelu tak nie jest.\vfill\ejct

```

UWAGI

1–19: Opisano sposób wykorzystania komendy `\item` do składu scenariuszy. (W przykładzie XIII przedstawiono inny sposób składu scenariusza.) Scenariusz napisany jest w języku hiszpańskim; łatwo zauważyć, jak składa się niezwykłą interpunkcję i symbole akcentowane.

3: `\-` opcjonalne dzielenie, sugestia podziału wyrazu (patrz też str. 22). Omawianą komendę umieszczono w tym miejscu, ponieważ T_EX miał problemy z poprawnym podziałem wiersza. `\’a` składa znak á.

4: Znaki `!‘` są interpretowane przez program jako pojedynczy znak ¡. Wiersz 18 zawiera instrukcję składu znaku ¿. Omówienie ligatur znajduje się w dodatku.

6, 7, itd.: Akcent nad i złożony jest w szczególności sposób: najpierw usuwa się kropkę, stosując komendę `\i`, która składa ı. Wtedy jak zwykle dokłada się akcent,

np. komendą `\’`.

13, 14: `\llap` przesuwają w lewo (w tym przypadku umieszcza na marginesie) wyrażenie w nawiasach. Komenda ta jest nazywana komendą „nakładania” (*overlap command*), ponieważ może być użyta do nakładania tekstu na tekst. `\heartsuit` składa \heartsuit . `\rlap` działa jak `\llap`, ale przesuwają tekst w prawo. Obydwie komendy są szczegółowo omówione w dodatku. `\line` wykorzystana jest do składu danego fragmentu tekstu w jednym wierszu.

25, 30, 32: `\"` składa akcent np. Ö.

28: Kolejna zmiana nagłówka strony `\headline`; należy pamiętać, że `\newpageno` jest specjalnie zdefiniowaną komendą i nie może być użyta bez definicji ze str. 2.

A: ¡Pues muchas gracias, amable caballero!

B: Sé que suena como una locura, pero trabajar contigo en estas cosas creativas me hace sentir como un potro salvaje. Uno pensaría que tanto trabajar de seso apagaría cualquier deseo físico, pero en realidad hace mucho tiempo que no me sentía así.

A: Si quieres que te diga la verdad, yo también.

B: Mira la puesta de sol, es igual que el poster que compramos aquella vez. Y mira el agua.

A: (Corriendo) Vamos.



.....



B: ¡Jolín! Nunca había dormido tan bien.

A: Lo mismo digo yo. Es estupendo despertat y estar realmente despierta, no solamente “despierta de café”.

B: ¿Donde habíamos quedado ayer, antes de que perdiéramos la cabeza y nos olvidáramos por completo de las matemáticas?

V. SPULAPKOWANE POWIERZCHNIE W PRZESTRZENI GÖDLA

Macierze; justowanie równań; indeksy; pochodne cząstkowe; przypisy.

W 1949 roku Kurt Gödel przedstawił kosmologiczne rozwiązanie równania Einsteina z pewnymi bardzo dziwnymi własnościami¹⁰. Wydaje się, że Gödel odkrył to rozwiązanie w trakcie badania idealistycznej koncepcji czasu, tj. problemu, czy „rzeczywistość składa się z nieskończonej liczby warstw teraźniejszości”. W jego modelu tak nie jest.

1 Po pierwsze, przez każdy punkt rzeczywistości przebiega zamknięta
2 krzywa typu czasowego. Po drugie, jest ona jednorodna. Z tych własności
3 wynika, że model nie posiada żadnej czasopodobnej hiperprzestrzeni.
4 „Rzeczywistość” w tym przypadku nie zawiera ani {it jednej/} warstwy
5 „teraźniejszości”.

6
7 Kilka własności rozwiązania G\`odla omówiono w pozycji 6; wykorzystamy
8 konwencję i notację tam zawartą. Dziedziną, w której jest zdefiniowana miara
9 G\`odla, jest $I \setminus \{R^4\}$. Współrzędne (t', x, y, z) mogą być wybrane tak, by
10 każda współrzędna $\in (-\infty, \infty)$ z miarą (odległością) daną przez
11 $ds^2 = -dt'^2 + dx^2 - (1/2) \exp\{2\sqrt{2}\omega x\} dy^2 + dz^2 - 2 \exp\{\sqrt{2}\omega x\} dt' dy$,
12 gdzie $\omega = \text{stała} > 0$. Ta miara spełnia równanie
13 $G_{ab} + \Lambda g_{ab} = 8\pi T_{ab}$, $\text{quad} \text{oraz} \text{quad} T_{ab} = \text{varrho}$
14 $U_a U_b$,
15 jeśli $(\partial_t)^a = U^a$ i $\omega^2 = -\Lambda = 4\pi \text{varrho}$.

17
18 Rozwiązanie G\`odla ma jeszcze jedną interesującą własność: istnieje oparta na
19 nim czasoprzestrzeń, zawierająca „boczenie spułapkowane powierzchni” (it
20 marginally trapped surfaces). Pojęcie spułapkowanej powierzchni zostało
21 wprowadzone w 1965 roku przez Rogera Penrose, w celu scharakteryzowania układu
22 podlegającego kolapsowi grawitacyjnemu $\{11\}$. Może ona być zdefiniowana jako
23 zwarte czasopodobne dwie przestrzenie bez granicy, gdzie obydwa systemy
24 czasowo-zorientowanych zerowych geodetyk (linii geodezyjnych), które emanują z
25 niej ortogonalnie, mają dodatnią konwergencję ρ . Boczenie spułapkowana
26 powierzchnia jest definiowana podobnie, ale z $\rho \geq 0$. Stephen Hawking
27 odkrył, że spułapkowane powierzchnie występują również w kilku rozszerzonych
28 modelach kosmologicznych $\{12\}$. (Te modele mogą być traktowane jako podlegające
29 kolapsowi grawitacyjnemu w odwrotnym czasowo kierunku.) Intrygujący jest fakt,
30 iż nieskolapsowane czasoprzestrzenie, taka jak G\`odla, powinna również zawierać
31 taką strukturę.

32
33 Istnienie boczenie spułapkowanych powierzchni łatwiej jest zobaczyć, jeśli
34 zdefiniujemy $\{6\}$ nowe współrzędne (t, r, ϕ, z) , gdzie $-\infty < t < \infty$,
35 $0 \leq r < \infty$, $0 \leq \phi < 2\pi$, i $-\infty < z < \infty$, przez
36 następujące przekształcenia: `\vfileject`

UWAGI

KOMENTARZ

Znaki \$ w uwagach wskazują, które komendy działają wyłącznie w trybie matematycznym.

7: Wiązanie ~ zapobiega podzieleniu słów „pozycji” i „6”. Patrz str. 2, 22 i 26.

9: $\!$ składa mały ujemny odstęp w trybie matematycznym, tzn. cofa tekst. Zapis w tej linii prowadzi do częściowego nałożenia dużych liter I oraz R.

12: \exp składa antykwą funkcję *exp*. Funkcje tego typu są zwykle właśnie w taki sposób wprowadzane do równań. ω składa ω .

14, 16: Λ składa Λ , varrho składa ρ , π składa π , a ∂ składa ∂ . Słowa, które mają występować w normalnym trybie tekstowym, z regularnymi odstępami tekstowymi, należy umieszczać w `\hbox`. Symbol `\`, składa małą spację.

26: ρ składa ρ , a \geq składa \geq .

35: Podział wiersza lub łamanie strony jest dozwolone w środku wzoru na znaku \leq . (Symbol relacji jest jednym z możliwych miejsc podziału wzorów; patrz do-

datek, matematyka: podział wierszy.) Aby zabezpieczyć się przed niezręcznym złamaniem strony, wzór został umieszczony w `\hbox`. Zawartość poziomego pudełka nigdy nie jest dzielona na dwa wiersze. W tym przypadku całe pudełko zostało przeniesione do następnego wiersza. ϕ składa ϕ , ∞ składa ∞ , a \leq składa \leq .

JUSTOWANIE RÓWNAŃ

Na str. 40 użyta zostanie komenda `\eqalignno`, która pozwala na wyrównanie układu równań oraz ich numerowanie. Struktura komendy odpowiada ogólnemu szablonowi justowania w $\text{T}_{\text{E}}\text{X}$ -u: wpisy są separowane znakiem `&`, a wiersze kończone komendą `\cr`. Równania na następnej stronie są wyrównane względem znaku `=`, przy czym zapisy po lewej stronie tego znaku są wyrównane prawostronnie, po prawej zaś — lewostronnie. Ostatni wpis w wierszu (wpis opcjonalny) jest numerem równania. Taki sposób eksponowania równań jest automatycznie kontynuowany po złamaniu strony.

Po pierwsze, przez każdy punkt rzeczywistości przebiega zamknięta krzywa typu czasowego. Po drugie, jest ona jednospójna. Z tych własności wynika, że model nie posiada żadnej czasopodobnej hiperprzestrzeni. „Rzeczywistość” w tym przypadku nie zawiera ani *jednej* warstwy „teraźniejszości”.

Kilka własności rozwiązania Gödla omówiono w pozycji 6; wykorzystamy konwencję i notację tam zawartą. Dziedziną, w której jest zdefiniowana miara Gödla, jest \mathbb{R}^4 . Współrzędne (t', x, y, z) mogą być wybrane tak, by każda współrzędna $\in (-\infty, \infty)$ z miarą (odległością) daną przez

$$ds^2 = -dt'^2 + dx^2 - (1/2) \exp(2\sqrt{2}\omega x) dy^2 + dz^2 - 2 \exp(\sqrt{2}\omega x) dt' dy,$$

gdzie $\omega = \text{stała} > 0$. Ta miara spełnia równanie

$$G_{ab} + \Lambda g_{ab} = 8\pi T_{ab}, \quad \text{oraz} \quad T_{ab} = \rho U_a U_b,$$

jeśli $(\frac{\partial}{\partial t'})^a = U^a$ i $\omega^2 = -\Lambda = 4\pi\rho$.

Rozwiązanie Gödla ma jeszcze jedną interesującą własność: istnieje oparta na nim czasoprzestrzeń, zawierająca „bocznie spułapkowane powierzchnie” (*marginally trapped surfaces*). Pojęcie spułapkowanej powierzchni zostało wprowadzone w 1965 roku przez Rogera Penrose, w celu scharakteryzowania układu podlegającego kolapsowi grawitacyjnemu¹¹. Może ona być zdefiniowana jako zwarte czasopodobne dwie przestrzenie bez granicy, gdzie obydwa systemy czasowo-zorientowanych zerowych geodetyk (linii geodezyjnych), które emanują z niej ortogonalnie, mają dodatnią konwergencję ρ . Bocznie spułapkowana powierzchnia jest definiowana podobnie, ale z $\rho \geq 0$. Stephen Hawking odkrył, że spułapkowane powierzchnie występują również w kilku rozszerzonych modelach kosmologicznych¹². (Te modele mogą być traktowane jako podlegające kolapsowi grawitacyjnemu w odwrotnym czasowo kierunku.) Intrygujący jest fakt, iż nieskolapsowana czasoprzestrzeń, taka jak Gödla, powinna również zawierać taką strukturę.

Istnienie bocznie spułapkowanych powierzchni łatwiej jest zobaczyć, jeśli zdefiniujemy⁶ nowe współrzędne (t, r, ϕ, z) , gdzie $-\infty < t < \infty$, $0 \leq r < \infty$, $0 \leq \phi \leq 2\pi$, i $-\infty < z < \infty$, przez następujące przekształcenia:

```

1  $$\eqalignno{\exp{(\sqrt{2}\omega x)} &= \cosh{2r}+\cos\phi,\sinh{2r}, &(1a)\cr
2  \omega y\exp{(\sqrt{2}\omega x)} &= \sin\phi,\sinh{2r}, &(1b)\cr
3  \textstyle{1\over 2}(\phi+\omega t-(\sqrt{2})t') &= \exp(-2r)\cr
4  \scriptstyle{1\over 2}\phi, &(1c)\cr
5  \noalign{\hbox{oraz}}\cr
6  z&=z.&(1d)\cr}$$
7  W tych współrzędnych odległość  $g_{ab}$  ma składniki określone przez
8   $g_{\mu\nu}=\{2\over\omega^2\}$ 
9   $\begin{pmatrix} -1&0&\sqrt{2}\sinh^2r \\ 0&1&0 \\ \sqrt{2}\sinh^2r &-\sinh^2r &(\sinh^2r-1) \end{pmatrix}$ 
10  $0&0&0$ 
11  $\sqrt{2}\sinh^2r &-\sinh^2r &(\sinh^2r-1)$ 
12  $0&0&0$ 
13  $\omega^2\over 2}$ 
14  $\left[ \begin{matrix} -(1-\sinh^2r) \\ \sqrt{2}\sinh^2r \\ \cosh^2r \end{matrix} \right]$ 
15  $\omega^2\over 2}$ 
16  $\left[ \begin{matrix} -(1-\sinh^2r) \\ \sqrt{2}\sinh^2r \\ \cosh^2r \end{matrix} \right]$ 
17  $\omega^2\over 2}$ 
18  $\left[ \begin{matrix} -(1-\sinh^2r) \\ \sqrt{2}\sinh^2r \\ \cosh^2r \end{matrix} \right]$ 
19  $\omega^2\over 2}$ 
20  $\left[ \begin{matrix} -(1-\sinh^2r) \\ \sqrt{2}\sinh^2r \\ \cosh^2r \end{matrix} \right]$ 
21  $\left[ \begin{matrix} -(1-\sinh^2r) \\ \sqrt{2}\sinh^2r \\ \cosh^2r \end{matrix} \right]$ 
22  $\left[ \begin{matrix} -(1-\sinh^2r) \\ \sqrt{2}\sinh^2r \\ \cosh^2r \end{matrix} \right]$ 
23  $\left[ \begin{matrix} -(1-\sinh^2r) \\ \sqrt{2}\sinh^2r \\ \cosh^2r \end{matrix} \right]$ 
24 są
25  $\Gamma^t_{tr}=2\text{tgh } r$ 
26  $\Gamma^r_{t\phi}=\sqrt{2}\sinh r\cosh r$ 
27  $\Gamma^{\phi}_{tr}=\sqrt{2}\over\sinh r\cosh r$ 
28  $\Gamma^t_{r\phi}=\sqrt{2}\text{tgh } r\sinh^2r$ 
29  $\Gamma^r_{\phi\phi}=\sinh r\cosh r(2\sinh^2r-1)$ 
30  $\Gamma^{\phi}_{r\phi}=2\text{tgh } r$ 
31  $\Gamma^{\phi}_{\phi\phi}=2\text{tgh } r$ 
32 Na podstawie tych równań łatwo jest sprawdzić, że krzywa zamknięta dana przez
33  $t=\text{const}$ ,  $r=\log(1+\sqrt{2})$ ,  $\phi=\phi(p)$ 
34  $z=\text{const}$ 

```

UWAGI

KOMENTARZ

Tu znajduje się główne omówienie justowania równań za pomocą komendy `\eqalignno`.

1, 2, 3, itd.: `\cosh`, `\sinh` itd. wykorzystuje się do wprowadzenia odpowiednich funkcji złożonych antykwa; `\,` dopasowuje odstęp.

3–4: Zauważ, jak `\textstyle` i `\scriptstyle` wpływają na rozmiar ułameków.

5: `\noalign` wyłącza z wyrównania część tekstu, tutaj `\hbox` zawierający słowo „oraz”.

8–12: `\mu` i `\nu` składają μ i ν . `\pmatrix` składa macierz w dużych nawiasach. Struktura komendy jest podobna do komendy wyrównującej równania: elementy są separowane znakami `&`, a komenda `\cr` oznacza koniec wiersza. Większość innych komend, pojawiających się w tych wierszach, była dyskutowana wcześniej. `\!` wykorzystana jest do umieszczenia wąskiej spacji. Komenda `\ast` składa znak `*`, który jest umieszczony jako indeks górny (zaraz będzie o tym mowa). `\eqno` pozwala określić numer równania.

13: `\vfootnote` składa ruchomy przypis, tzn. nie związany z jakąś określoną częścią głównego dokumentu. Symbol przypisu w tym przypadku to `*`; uzyskujemy efekt przypisu do macierzy.

17–19: Macierz w nawiasach kwadratowych otrzy-

mano za pomocą komendy `\matrix` wraz z nawiasami ograniczającymi zawartość. Komendy `\left` i `\right` powodują, że nawiasy są dopasowane wielkością do eksponowanego wyrażenia. Inne komendy w tym miejscu są albo komendami składającymi antykwę nazwy funkcji, albo komendami odstępów, albo literami greckimi.

21: `\Gamma` składa `\Gamma`. `\phantom` wstawia dodatkowy odstęp, odpowiadający szerokości tekstu znajdującego się w nawiasach. W tym przypadku we frakcji dolnej znalazł się odstęp o rozmiarze znaku „a” i dwa już widoczne symbole. Po co taki trik? Porównaj tak złożony symbol Γ z jednym ze znajdujących się poniżej, np. w wierszu 29. `\textstyle` składa ułamek mniejszy niż w trybie eksponowanym.

25–30: `\eqalign` jest jeszcze jedną komendą justującą. Nie pozwala ona na numerowanie pojedynczych równań ani na kontynuowanie wyrównania na następnej stronie. Umożliwia natomiast wielokrotne wyrównanie w obrębie tego samego wyrażenia eksponowanego. Każde wyrównanie jest składane osobno, przy użyciu `&` i `\cr`, całość umieszczana jest między podwójnymi znakami `$$`. Tutaj dwa zbiory justowanych równań, umieszczone obok siebie, oddzielone są odstępem poziomym `\quad`. Pozostałe komendy, znajdujące się w tych wierszach, były już objaśniane.

$$\exp(\sqrt{2}\omega x) = \cosh 2r + \cos \phi \sinh 2r, \quad (1a)$$

$$\omega y \exp(\sqrt{2}\omega x) = \sin \phi \sinh 2r, \quad (1b)$$

$$\operatorname{tg} \frac{1}{2}(\phi + \omega t - (\sqrt{2})t') = \exp(-2r) \operatorname{tg} \frac{1}{2}\phi, \quad (1c)$$

oraz

$$z = z. \quad (1d)$$

W tych współrzędnych odległość g_{ab} ma składniki określone przez

$$g_{\mu\nu} = \frac{2}{\omega^2} \begin{pmatrix} -1 & 0 & \sqrt{2} \sinh^2 r & 0 \\ 0 & 1 & 0 & 0 \\ \sqrt{2} \sinh^2 r & 0 & -\sinh^2 r (\sinh^2 r - 1) & 0 \\ 0 & 0 & 0 & \frac{\omega^2}{2} \end{pmatrix}^*, \quad (2a)$$

macierz odwrotna zaś ma elementy

$$g^{\mu\nu} = \frac{\omega^2}{2} \begin{bmatrix} -(1 - \sinh^2 r) / \cosh^2 r & 0 & \sqrt{2} / \cosh^2 r & 0 \\ 0 & 1 & 0 & 0 \\ \sqrt{2} / \cosh^2 r & 0 & 1 / \sinh^2 r \cosh^2 r & 0 \\ 0 & 0 & 0 & 2 / \omega^2 \end{bmatrix}. \quad (2b)$$

Wtedy jedynymi niezerowymi wartościami

$$\Gamma_{\mu\nu}^{\lambda} = \frac{1}{2} g^{\lambda\rho} \{ \partial_{\nu} g_{\rho\mu} + \partial_{\mu} g_{\rho\nu} - \partial_{\rho} g_{\mu\nu} \}$$

są

$$\begin{aligned} \Gamma_{tr}^t &= 2 \operatorname{tgh} r, & \Gamma_{r\phi}^t &= \sqrt{2} \operatorname{tgh} r \sinh^2 r, \\ \Gamma_{t\phi}^r &= \sqrt{2} \sinh r \cosh r, & \Gamma_{\phi\phi}^r &= \sinh r \cosh r (2 \sinh^2 r - 1), \\ \Gamma_{tr}^{\phi} &= \frac{\sqrt{2}}{\sinh r \cosh r}, & \Gamma_{r\phi}^{\phi} &= 2 \operatorname{tgh} r. \end{aligned}$$

Na podstawie tych równań łatwo jest sprawdzić, że krzywa zamknięta dana przez

$$t = \operatorname{const}, \quad r = \log(1 + \sqrt{2}), \quad \phi = \phi(p) \quad \text{i} \quad z = \operatorname{const} \quad (3)$$

* Tj. $ds^2 = \frac{2}{\omega^2} (-dt^2 + dr^2 - \sinh^2 r (\sinh^2 r - 1) d\phi^2 + 2\sqrt{2} \sinh^2 r d\phi dt) + dz^2$.


```

1 \noindent jest niegeodezyjną krzywą zerową, której przyspieszenie  $a^b$ 
2 ( $=N^c\nabla\!_cN^b$ , gdzie  $N^b=(\frac{\partial}{\partial p})^b$ ,
3  $a^b\nabla\!_c$  jest pochodną kowariantną) ma składniki określone przez
4 
$$\text{\centerline}\{(0, \, \sinh r \cosh r, (2 \sinh^2 r - 1) \dot{\phi}^2, \, 0, \, 0)\}$$
,
5 gdzie  $\dot{\phi}$  dotyczy pochodnej w odniesieniu do  $p$ . W obszarze
6  $r < \log(1 + \sqrt{2})$   $du =$  powierzchnia ( $it two = surface$ ), dana przez
7  $t = const$ ,  $r = const$ , będzie typu przestrzennego. Te powierzchnie mogą być
8 kompaktyfikowane przez identyfikację np.  $z=0$  z  $z=1$ . Otrzymana
9  $du =$  powierzchnia będzie miała topologię  $S^1 \times S^1$ . Składniki stycznej do
10 dwóch układów zerowych geodetyk, które emanują ortogonalnie z tych powierzchni,
11 można zapisać jako
12 
$$N^t_{\pm} = \{1 - \frac{\sinh^2 r}{\cosh^2 r}, \cr$$

13 
$$N^r_{\pm} = \pm \frac{(1 - \sinh^2 r)^{1/2}}{\cosh^2 r}, \cr$$

14 
$$N^{\phi}_{\pm} = \frac{-\sqrt{2}}{\cosh^2 r}, \cr$$

15 
$$N^z_{\pm} = 0. \cr \}$$
eqno(4)
16 A konwergencję tych dwóch układów można obliczyć następująco
17 
$$\rho_{\pm} \equiv -\nabla_a N^a_{\pm} = \frac{-1}{\sqrt{-g}}$$

18 
$$\frac{\partial}{\partial x^{\mu}} N^{\mu}_{\pm} \over \partial x^{\mu} \cr$$

19 
$$= \mp \left[ \frac{2 \sinh^2 r - 1}{\sinh r \sqrt{1 - \sinh^2 r}} \right], \cr \}$$

20 gdzie  $x^{\mu}$  oznacza współrzędne. Wtedy dla  $r$  danego równaniem
21  $\sinh^2 r = 1/2$  obydwa układy mają zerową konwergencję. To jest właśnie bocznie
22 spułapkowana powierzchnia.
23 \vfil\ejct

```

UWAGI

KOMENTARZ

Znaki $\$$ w uwagach wskazują, które komendy działają wyłącznie w trybie matematycznym.

2–3: $\$ \nabla$ składa znak ∇ , $\$ \partial$ składa ujemny odstęp (przesuwa materiał trochę wstecz).

4: W wierszu tym znalazło się kilka komend, dotyczących wąskich odstępów, które umieszczono m.in. przed nawiasami, by zapobiec zbytniemu ściśnięciu tekstu.

4, 5: $\$ \dot{\phi}$ umieszcza kropkę nad tym, co następuje po nazwie komendy.

12–15: Cały zbiór równań, które są wyrównane komendą $\$ \text{align}$, może być numerowany za pomocą $\$ \text{eqno}$, w przeciwieństwie do pojedynczych równań. Aby numerować każde równanie osobno, należy uciec się do trików (np. wprowadzić osobną kolumnę numerów równań i dostosować odstępy w celu odpowiedniego wyrównania kolumn).

17–19: $\$ \rho$ składa ρ , $\$ \pm$ składa \pm , $\$ \equiv$ składa \equiv , $\$ \nabla$ składa ∇ , $\$ \partial$ składa ∂ , $\$ \mu$ składa μ , $\$ \mp$ składa \mp a $\$ \sinh$ składa „sinh”. Metoda $\$ \left$ i $\$ \right$, zastosowana tu w odniesieniu do nawiasów kwadratowych, działa z dowolnymi ogranicznikami wewnątrz wyrażenia eksponowanego (patrz dodatek, matematyka: ograniczniki).

WSTĘPNE UWAGI DOTYCZĄCE TABEL

Proste tabele łatwo składa się za pomocą \TeX -a, ale skomplikowane, zwłaszcza obramowane, mogą sprawiać trudności. Na kilku następnych stronach przedstawiono podstawowe zasady tworzenia tabel. Konstruowanie tabel za każdym razem od początku nie jest zalecane, szczególnie jeśli często z nich korzystamy. Proponuje się definiowanie komend, które zastąpią część tej mozolnej pracy. Pod hasłem *tabele* w dodatku znajdują się przykłady komend formatujących tabele; można je skopiować i wykorzystywać. Jest także kilka gotowych pakietów, rozpowszechnianych przez grupy zajmujące się \TeX -em, umożliwiających tworzenie wymyślnych tabel przy małym nakładzie pracy.

$\$ \text{halign}$ jest komendą pozwalającą na wyrównywanie w \TeX -u. Tabele można tworzyć bezpośrednio przez zastosowanie tej komendy lub za pomocą innych komend na niej opartych, np. $\$ \text{settabs}$. $\$ \text{halign}$ wymaga wpisania preambuły, krótkiej przykładowej linii regulującej resztę tabeli. $\$ \text{settabs}$ wymaga albo wyraźnej specyfikacji liczby kolumn (tworzy wtedy kolumny równej szerokości), albo podania przykładowej linii, zawierającej wpisy o szerokości określającej żądane szerokości kolumn.

Wszystkie te opcje są przedstawione dalej.

jest niegeodezyjną krzywą zerową, której przyspieszenie $a^b (= N^c \nabla_c N^b$, gdzie $N^b = (\frac{\partial}{\partial p})^b$, a ∇_c jest pochodną kowariantną) ma składniki określone przez

$$(0, \sinh r \cosh r (2 \sinh^2 r - 1) \dot{\phi}^2, 0, 0),$$

gdzie „ $\dot{}$ ” dotyczy pochodnej w odniesieniu do p . W obszarze $r < \log(1 + \sqrt{2})$ dwu-powierzchnia (*two-surface*), dana przez $t = \text{const}$, $r = \text{const}$, będzie typu przestrzennego. Te powierzchnie mogą być kompaktyfikowane przez identyfikację np. $z = 0$ z $z = 1$. Otrzymana dwu-powierzchnia będzie miała topologię $S^1 \times S^1$. Składniki stycznnej do dwóch układów zerowych geodetyk, które emanują ortogonalnie z tych powierzchni, można zapisać jako

$$\begin{aligned} N_{\pm}^t &= \frac{1 - \sinh^2 r}{\cosh^2 r}, \\ N_{\pm}^r &= \pm \frac{(1 - \sinh^2 r)^{1/2}}{\cosh^2 r}, \\ N_{\pm}^{\phi} &= \frac{-\sqrt{2}}{\cosh^2 r}, \\ N_{\pm}^z &= 0. \end{aligned} \tag{4}$$

A konwergencję tych dwóch układów można obliczyć następująco

$$\begin{aligned} \rho_{\pm} &\equiv -\nabla_a N_{\pm}^a = \frac{-1}{\sqrt{-g}} \frac{\partial(\sqrt{-g} N_{\pm}^{\mu})}{\partial x^{\mu}} \\ &= \mp \left[\frac{2 \sinh^2 r - 1}{\sinh r \sqrt{1 - \sinh^2 r}} \right], \end{aligned}$$

gdzie x^{μ} oznacza współrzędne. Wtedy dla r danego równaniem $\sinh^2 r = 1/2$ obydwa układy mają zerową konwergencję. To jest właśnie bocznie spulapkowana powierzchnia.

```

1 \centerline{\bf VI. SUNIL GAVASKAR: A STATISTICAL TRIBUTE%
2 \footnote{\rm This section contains unexplained references to
3 cricket. Those who follow the game will know what they mean; those who do
4 not are unlikely to care. The statistics are correct up to 1987.}}
5 \centerline{\sl Tabele.}
6 \headline{\hfil\it Przykład VI\newpage}
7 \bigskip
8 \noindent Towards the end of 1983, the Indian batsman Sunil Gavaskar scored
9 his 8,115th Test run to become the highest scorer in the history of
10 international cricket. He was averaging at the time a little over 50 runs for
11 every completed inning. Before the year was over he had also made his 30th
12 Test century, breaking Bradman's record of 29 set in 1948.
13
14 Gavaskar was then 34. He had been in international cricket for 13 years,
15 and he had played 99 times for India in this period. He had consistently
16 opened the batting and had faced some of the fastest bowlers in the
17 history of the game. The West Indian pace attack, in particular, had been
18 exceptionally fierce over most of the period that he had played. And against
19 them his record had been particularly good:
20 \bigskip
21 \vbox{\baselineskip=16 true pt\settabs7\columns
22 \+&&&\hfil{\bf Table 1}&\cr
23 \+&&\it Tests&\it Runs&\it Avg.&\it 100s\cr
24 \+&&\sl 27&\sl 2749&\sl 65.45&\sl 13\cr}
25 \bigskip
26 \noindent After this arduous and highly successful career, it seemed
27 reasonable to suppose that he would want to retire.
28
29 For a while, it did look as though he had lost some of his determination.
30 A series of low scores in 1984-85 followed his successes of 1983:
31 \bigskip
32 \vbox{\baselineskip=15pt
33 \halign{\hfil\sl#&\quad\hfil\it#\hfil&\quad#\hfil\cr
34 \multispan3\hfil\bf Table 2\hfil\cr
35 Country&Year&Scores:\cr Pakistan&1984&48, 37, 35.\cr
36 England&1984--85&27, 5, 1, 65, 13, 17, 3, 9.\cr}}

```

UWAGI

KOMENTARZ

(Tekst tego przykładu pozostawiono w oryginale — przyp. tłum.) Wstępne uwagi dotyczące tabel znajdują się na str. 42.

1: Znak % likwiduje pojedynczą spację, którą T_EX automatycznie umieszcza na końcu wiersza wczytywanego pliku.

2: `\footnote` wywołane jest bez problemu w ramach `\centerline`.

9, 11: Ponieważ frakcja górna jest częścią trybu matematycznego, cokolwiek umieści się w tym miejscu, będzie złożone kursywą matematyczną (można także wybrać inny krój). Zwróć uwagę na frakcję górną *th*.

21–24: `\vbox` zapewnia, że dany materiał (to co następuje zaraz po nazwie komendy i jest ograniczone klamrami) nie zostanie podzielony na dwie strony. W celu dopasowania odstępów, wewnątrz pudełka można zmienić lokalnie wartość `\baselineskip`; nie będzie to miało wpływu na skład pozostałego tekstu. Za pomocą komendy `\settabs` utworzono tabelę składającą się z 7 kolumn, z tym że 3 kolumny (dwie pierwsze i ostatnia) pozostawiono puste, aby zmniejszyć szerokość kolumn, gdyż `\settabs` składa tabelę na całą szerokość tekstu, z kolumnami o jednakowej szerokości. Daje to również przyjemny efekt uboczny,

wyśrodkowania tabeli. Każdy kolejny wiersz musi zaczynać się znakiem `\+`, a kończyć komendą `\cr`. Tak jak i we wszystkich innych wyrównaniach w T_EX-u, znak `&` przenosi nas z kolumny do kolumny, a komenda `\cr` oznacza koniec wiersza. Standardowo poszczególne wpisy są wyrównywane do lewej. Tytuł tabeli jest w rzeczywistości dosuniętym do prawej (za pomocą komendy `\hfil`) wpisem w czwartej kolumnie. **33–36:** W strukturze komendy `\halign` pierwszy wiersz jest preambułą, w której można określić krój pisma, rozmieszczenie tekstu. W tym przypadku pierwsza kolumna jest wyrównana do prawej i wpisy w niej są składane krojem pochyłym (*slanted*), druga wyrównana do lewej — domyślnym krojem antykwy. `\quad` umieszcza odstępy między kolumnami. Szerokość kolumn jest zmienna, uzależniona od najdłuższego wpisu w danej kolumnie.

34: `\multispan` łączy wyspecyfikowaną liczbę kolumn w jedną, np. w celu umieszczenia tytułu tabeli. Komendy dotyczące krojów pisma, wprowadzone bezpośrednio przy wpisach, mają pierwszeństwo przed komendami z preambuły, tak samo jak komendy dotyczące odstępów (np. `\hfill` zadziała silniej niż `\hfil`).

VI. SUNIL GAVASKAR: A STATISTICAL TRIBUTE‡

Tabele.

Towards the end of 1983, the Indian batsman Sunil Gavaskar scored his 8,115th Test run to become the highest scorer in the history of international cricket. He was averaging at the time a little over 50 runs for every completed inning. Before the year was over he had also made his 30th Test century, breaking Bradman's record of 29 set in 1948.

Gavaskar was then 34. He had been in international cricket for 13 years, and he had played 99 times for India in this period. He had consistently opened the batting and had faced some of the fastest bowlers in the history of the game. The West Indian pace attack, in particular, had been exceptionally fierce over most of the period that he had played. And against them his record had been particularly good:

Table 1

<i>Tests</i>	<i>Runs</i>	<i>Avg.</i>	<i>100s</i>
27	2749	65.45	13

After this arduous and highly successful career, it seemed reasonable to suppose that he would want to retire.

For a while, it did look as though he had lost some of his determination. A series of low scores in 1984-85 followed his successes of 1983:

Table 2

<i>Country</i>	<i>Year</i>	<i>Scores:</i>
<i>Pakistan</i>	<i>1984</i>	48, 37, 35.
<i>England</i>	<i>1984-85</i>	27, 5, 1, 65, 13, 17, 3, 9.

‡ This section contains unexplained references to cricket. Those who follow the game will know what they mean; those who do not are unlikely to care. The statistics are correct up to 1987.

```

1 \bigskip \noindent
2 In seven Test matches he had made 260 runs at an average of 23.64. In late 1985,
3 he asked to be dropped in the batting order against Sri Lanka and made 51, 0,
4 52, 19, 49 and 15 not out. It seemed as if his career must surely be over.
5
6 But Gavaskar had gone through lean periods before, and he had always been able
7 to pull himself out. He did so once more. Returning as opener against Australia
8 in 1985-86, he made 166 not out, 6, 8, and 172. He subsequently scored two
9 more centuries, continued to maintain an overall average of 50 and, in early
10 1987, he became the first player to cross 10,000 Test runs.
11
12 Though it is clear that these are exceptional achievements, their full magnitude
13 may not immediately be apparent. By 1971, the year Gavaskar started playing
14 Test cricket, over 15 international players (from England, Australia and the
15 West Indies) had accumulated career run aggregates of 4,500 or more, whereas
16 only 3 Indian batsmen (Umrigar, Manjrekar and Borde) had scored more than 3,000.
17 Umrigar, at the top of the Indian list, had a total run aggregate that was
18 almost exactly half the international record (see Table 3). He was also the only
19 Indian batsman who had made more than 10 centuries. No Indian batsman who had
20 played for any real length of time had averaged over 50.
21 \midinsert
22 Ta wstawka umieszczona jest na następnym stronie.
23 \endinsert
24 In such circumstances, not much was seriously expected from Indian batsmen.
25 I went to the same high school that Gavaskar did (though I was many years his
26 junior and did not know him, except by his reputation as a successful member
27 of the Indian schoolboy's team), and it would have been considered the purest
28 fantasy there had anyone suggested that an Indian batsman would one day
29 score as many runs as Hammond or as many centuries as Bradman. It is against
30 a background of low expectations, and of satisfaction with middle-level
31 success, that Gavaskar has played out most of his~career.
32 \vfil\ejct

```

UWAGI

1: Komenda `\bigskip` w tym miejscu jest częścią głównego pliku pomiędzy tabelą (na str. 44) a następującym dalej tekstem. \TeX akurat w tym miejscu złamał stronę, więc nie ma żadnego dodatkowego odstępu u góry strony.

21–23: `\midinsert` próbuje umieścić materiał akurat w miejscu wystąpienia danej komendy. W razie niepowodzenia, tak jak w tym miejscu, umieszcza dany materiał u góry następnej strony. Inny typ wstawienia, `\topinsert`, umieszcza materiał u góry pierwszej wolnej strony. Koniec wstawek oznacza się komendą `\endinsert`.

31: Wiązanie `~` wykorzystane jest w celu zabezpieczenia przed izolacją słowa w ostatniej linii. Patrz str. 2, 22 i 26.

UWAGI NA TEMAT OBRAMOWANYCH TABEL
Skład kolejnej tabeli nie różni się zasadniczo od drugiej tabeli na poprzedniej stronie, nowością jest obecność kresek (*rules*). `\offinterlineskip` likwiduje odstępy między wierszami, a `\strut` zapobiega stykaniu się wierszy. Jest to niewidoczny stempel (podpora), o zerowej szerokości, ale niezerowej wysokości i odsadce, który odsuwa od siebie wiersze na stałą odległość. Pionowe kreski w tabeli zdefiniowane są w preambule jako wpisy w tabeli; kreski poziome (tak jak każdy inny materiał, który musi ignorować wyrównywanie w kolumnach) zadane są za pomocą komendy `\noalign`. Inną przydatną komendą jest `\omit`, która umieszczona na początku dowolnego wpisu ma pierwszeństwo przed ustawieniami z preambuły. Pozwala to na wybór innego sposobu formatowania.

In seven Test matches he had made 260 runs at an average of 23.64. In late 1985, he asked to be dropped in the batting order against Sri Lanka and made 51, 0, 52, 19, 49 and 15 not out. It seemed as if his career must surely be over.

But Gavaskar had gone through lean periods before, and he had always been able to pull himself out. He did so once more. Returning as opener against Australia in 1985-86, he made 166 not out, 6, 8, and 172. He subsequently scored two more centuries, continued to maintain an overall average of 50 and, in early 1987, he became the first player to cross 10,000 Test runs.

Though it is clear that these are exceptional achievements, their full magnitude may not immediately be apparent. By 1971, the year Gavaskar started playing Test cricket, over 15 international players (from England, Australia and the West Indies) had accumulated career run aggregates of 4,500 or more, whereas only 3 Indian batsmen (Umrigar, Manjrekar and Borde) had scored more than 3,000. Umrigar, at the top of the Indian list, had a total run aggregate that was almost exactly half the international record (see Table 3). He was also the only Indian batsman who had made more than 10 centuries. No Indian batsman who had played for any real length of time had averaged over 50.

In such circumstances, not much was seriously expected from Indian batsmen. I went to the same high school that Gavaskar did (though I was many years his junior and did not know him, except by his reputation as a successful member of the Indian schoolboy's team), and it would have been considered the purest fantasy there had anyone suggested that an Indian batsman would one day score as many runs as Hammond or as many centuries as Bradman. It is against a background of low expectations, and of satisfaction with middle-level success, that Gavaskar has played out most of his career.

KOMENTARZ

Ogólne uwagi na temat tabel znajdują się na str. 46. Tutaj zamieszczono jedynie plik wejściowy tabeli ze strony składu (zawartość wstawki między `\midinsert` i `\endinsert` na str. 46), nie dotyczy on tekstu.

2: Znaki `$$` oraz `\vbox` nie mają decydującego wpływu na skład tej tabeli, aczkolwiek są przydatne. Znaki `$$` włączają eksponowany tryb matematyczny oraz wyśrodkowują materiał (zamykająca para znajduje się w linii 37). W przypadku takiego eksponowanego mechanizmu konieczne jest użycie komendy `\vbox`, co ochrania wnętrze tabeli przed efektami trybu matematycznego i zabezpiecza materiał przed podziałem na strony.

3: Preambuła z `\halign` jest typu „otwartego”: rozpoczyna się znakiem `&`, który każe programowi powtarzać pewien wzorzec dowolną liczbę razy. Wpis w preambule postaci `A&& B& C\cr` będzie czytany przez program jako `A& B& C& B& C& B.` . . W takim przypadku liczba kolumn zostanie ustalona na podstawie wiersza

tabeli o największej liczbie elementów (komórek). Wpisy w tej tabeli to naprzemiennie pojawiające się pionowe kreski (`\vrule`) i tekst umieszczany w miejsce parametru `#`. Ostatnie elementy w wierszu poprzedzone są komendą `\strut`.

4: `\multispan` łączy wszystkie 13 kolumn (pionowe kreski są traktowane również jako kolumny), komendy `\hfil` wyśrodkowują tytuł tabeli.

5, 6, itd.: `\noalign` pozwala wprowadzić materiał, który ignoruje wyrównywanie z tabeli.

7, itd.: `\omit` daje możliwość ignorowania wzorca z preambuły, podobnie jak `\strut`. W tym przypadku dokładany jest 3-punktowy odstęp nad i pod każdą z poziomych kreszek.

13, 16, 22, 25: Wpis umieszczony w pierwszej kolumnie nie będącej pionową kreską, pozostałe kolumny są puste.

30, 34, itd.: W wierszach tych znajdują się dwa wpisy: pierwszy obejmujący 3 kolumny tabeli, drugi 10 kolumn; pomimo iż umieszczone są poza ramką tabeli, nadal stanowią część ogólnego wyrównania.

Table 3

Comment	Player	T	R	Avg.	100s
<i>Highest scorer till 1970</i>	W.R. Hammond (Eng.)	85	7249	58.45	22
<i>Highest Indian scorer till 1970</i>	P.R. Umrigar	59	3631	42.22	12
<i>Second highest scorer till 1987</i>	G. Boycott (Eng.)	108	8114	47.22	22
<i>Second highest Indian scorer till 1987</i>	G.R. Viswanath	91	6080	41.93	14

Sources: –*Playfair Cricket Annual*,
Queen Anne Press (1986).
–*The Encyclopaedia of Indian Cricket*,
L.N. Mathur, Rajhans Prakashan (1965).

Seen against this background, the extent of Gavaskar's achievement is all the more remarkable. Up to mid-1987, seven players had scored more than 7,000 runs in Test cricket. Apart from the two names in Table 3, they were Greg Chappell (7110) of Australia, Cowdrey (7624) of England and Lloyd (7515) and Sobers (8032) of the West Indies. And then, after Boycott, there was the 2,000-run jump to Gavaskar. Eight batsmen had made between 20 and 25 centuries, and then came Sobers (26), Bradman (29) of Australia and Gavaskar (34).

Partly of course, this success was due to the steadily rising standards in Indian cricket, and the steadily improving overall performances, since 1970. But that is not the complete story. In some ways, the better standards and resulting higher expectations existed *because* of the goals that Gavaskar had set. Also, a glance at last entry in Table 3 shows that the second most successful Indian batsman was a considerable distance behind. This gap between Gavaskar and the other Indian batsmen was dramatically seen in the last Test


```

1 Seen against this background, the extent of Gavaskar's achievement is all the
2 more remarkable. Up to mid-1987, seven players had scored more than 7,000 runs
3 in Test cricket. Apart from the two names in Table 3, they were Greg Chappell
4 (7110) of Australia, Cowdrey (7624) of England and Lloyd (7515) and Sobers
5 (8032) of the West Indies. And then, after Boycott, there was the 2,000-run jump
6 to Gavaskar. Eight batsmen had made between 20 and 25 centuries, and then came
7 Sobers (26), Bradman (29) of Australia and Gavaskar (34).
8
9 Partly of course, this success was due to the steadily rising standards in
10 Indian cricket, and the steadily improving overall performances, since 1970. But
11 that is not the complete story. In some ways, the better standards and resulting
12 higher expectations existed {\it because\} of the goals that Gavaskar had set.
13 Also, a glance at last entry in Table 3 shows that the second most successful
14 Indian batsman was a considerable distance behind. This gap between Gavaskar and
15 the other Indian batsmen was dramatically seen in the last Test match that he
16 played, against Pakistan in Bangalore in 1987. Set to make 221 to win on a
17 difficult wicket, India lost by 16 runs. Their second inning scorecard, shown in
18 Table 4, speaks for itself.\bigskip
19 \centerline{\bf Table 4}
20 \nobreak\medskip
21 \noindent\hfil \vbox{\baselineskip=13,5pt
22 \settabs\+Extras (22b, 51b)\quad&c Youssef b Tauseef\quad&ABC&\cr
23 \+S. Gavaskar &c Rizwan b Qasim &\hfill 96&\cr
24 \+K. Srikanth &lbw b Akram &\hfill 6&\cr
25 \+M. Amarnath &c Youssef b Akram &\hfill 0&\cr
26 \+D. Vengsarkar &b Tauseef &\hfill 19&\cr
27 \+K. More &lbw b Tauseef &\hfill 3&\cr
28 \+M. Azharuddin &c and b Qasim &\hfill 26&\cr
29 \+R. Shastri &c and b Qasim &\hfill 4&\cr
30 \+Kapil Dev &b Qasim &\hfill 2&\cr
31 \+R. Binny &c Youssef b Tauseef &\hfill 15&\cr
32 \+S. Yadav &b Tauseef &\hfill 4&\cr
33 \+M. Singh &\not out &\hfill 2&\cr
34 \+Extras (22b, 51b) & &\hfill 27&\cr
35 \+\bf Total & &\bf\hfill 204&\cr}
36 \vfil\ejct

```

UWAGI

Akapity znajdujące się u góry pliku wejściowego zostały wydrukowane na str. 49. Plik ten umieszczono tutaj ze względu na wolne miejsce. Tabela na tej stronie to jeden ze sposobów wykorzystania faktu, iż TeX ignoruje większość odstępów w pliku wejściowym. Wprowadzenie tabeli w ten sposób ułatwia orientację w poszczególnych wpisach.

19: Tytuł tabeli jest umieszczony poza pudełkiem `\vbox` zawierającym wyrównanie.

20: `\nobreak\medskip` zapobiega złamaniu strony w miejscu komendy `\medskip`, pomiędzy tytułem i tabelą.

21: Jednym ze sposobów wyśrodkowania tekstu jest umieszczenie przed nim komend `\noindent\hfil`. Kombinacja ta działa tylko wtedy, gdy umieszczona jest na początku akapitu. (Standardowy proces kończenia akapitów ma wbudowane wypełnienie `\hfil`, które równoważy te jawnie wprowadzone justunki.)

22–35: Kolejnym sposobem na skład tabel jest wykorzystanie komendy `\settabs`. Zamiast umieszczać liczbę kolumn, na początku wprowadzamy przykładowy wiersz, w którym szerokość każdego wpisu jest większa niż najszerszy wpis w danej kolumnie. Kolumny tabeli są dopasowane do tych wpisów. Struk-

tura ta jest podobna do wcześniej omawianego wyrównywania, ze znakiem `&` jako znacznikiem tabulacji i komendą `\cr` kończącą wiersze.

KOMENTARZ

W standardowym wyrównaniu w Plain TeX-u kolumny stykają się. Odległość między kolumnami określona jest za pomocą parametru zwanego `\tabskip`. Domyślna wartość tego parametru w Plain TeX-u wynosi `0pt`, ale może być dowolnie zmieniona. Aby nie zmienić standardowych ustawień tabel, należy pamiętać, by nowe ustawienia otoczyć klamrami, w celu ograniczenia ich działania tylko do danej grupy.

Przez wyspecyfikowanie zmiennego (elastycznego) rozmiaru `\tabskip`, np. `\tabskip=20pt plus 5pt minus 5pt`, można spowodować, by tabela skonstruowana przez `\halign` wypełniała zadaną szerokość składu. Jeśli ma on mieć szerokość np. 4 cali, należy wpisać `\halign to 4in{` i wtedy kontynuować jak wcześniej. Oczywiście nowe ustawienie `\tabskip` musi mieć wystarczającą zdolność rozciągania (liczba po słowie `plus`), aby odpowiednio wypełnić przestrzeń.

match that he played, against Pakistan in Bangalore in 1987. Set to make 221 to win on a difficult wicket, India lost by 16 runs. Their second inning scorecard, shown in Table 4, speaks for itself.

Table 4

S. Gavaskar	c Rizwan b Qasim	96
K. Srikanth	lbw b Akram	6
M. Amarnath	c Youssef b Akram	0
D. Vengsarkar	b Tauseef	19
K. More	lbw b Tauseef	3
M. Azharuddin	c and b Qasim	26
R. Shastri	c and b Qasim	4
Kapil Dev	b Qasim	2
R. Binny	c Youssef b Tauseef	15
S. Yadav	b Tauseef	4
M. Singh	not out	2
Extras (22b, 51b)		27
Total		204

```

1 \centerline{\bf VII. ZEROWE ROZWIĄZANIA RÓWNANIA  $\ddot{x} + F(t)x = 0$ }
2 \centerline{\sl Pochodne i całki; justowanie równań.}
3 \headline{\hfil\it Przykład VII\newpage}
4 \bigskip
5 \noindent Równanie  $\frac{d^2x}{dt^2} + F(t)x = 0$  \leqno{\rm(i)} opisuje
6 zachowanie się wielu różnych układów. Dla  $F = \text{const} > 0$  opisuje ono zwykły
7 oscylator harmoniczny. Opisuje także ruch cząstki naładowanej w~osiowo
8 symetrycznym polu elektrycznym. (Tutaj  $F$  nie może być stałą ani nawet liczbą
9 dodatnią13.) W~obu przypadkach interesują nas rozwiązania oscylatora, tzn.
10 rozwiązania, w~których  $x$  przebiega wielokrotnie przez 0. Innym równaniem tego
11 typu jest równanie ogniskowania geodetyki dla przyczynowych geodetyk
12 w~czasoprzestrzeni. Można dojść do niego przez wprowadzenie nowej zmiennej
13  $y = (\dot{x}/x)$ 14. Wtedy
14  $\dot{y} \equiv \frac{dy}{dt} = -y^2 - F(t)$ . \leqno{\rm(ii)}
15 To równanie ma formę równania ogniskowania
16 geodetyki14,15. Interesujące nas w~tym momencie rozwiązania (ze względu
17 na ilość zastosowań) otrzymujemy, gdy istnieją  $t_1$  i  $t_2$ , gdzie
18  $t_1 < t_2$ , takie że  $y \rightarrow -\infty$ , gdy  $t \rightarrow t_1^+$ , i  $y$ 
19  $\rightarrow -\infty$ , gdy  $t \rightarrow t_2^-$  (lub, równolegle, rozwiązania
20 równania (i) z dwoma zerami). Następujące wyniki ustalają warunek na
21 istnienie takiego rozwiązania15.
22
23 \proclaim Twierdzenie. Niech  $F(t)$  będzie funkcją ciągłą. Załóżmy, że
24 \item{\rm a)} istnieje dowolne  $t_0$ , takie że  $F(t_0) \neq 0$ ,
25 \item{\rm b)} istnieje dowolne  $b > 0$ , takie że dla dowolnych  $t_1 < t_0$ 
26 i  $t_2 > t_0$  istnieją przedziały domknięte  $I_1 \subset (t_1, t_0)$  (tj.  $\sup I_1 < t_0$ )
27 i  $I_2 \subset (t_0, t_2)$  (tj.  $\inf I_2 > t_0$ ) o~długości  $\geq b$ , gdzie
28  $\int_{I_1} |F(t)| dt \geq 0$  i  $\int_{I_2} |F(t)| dt \geq 0$  oraz
29  $\int_{I_1} |F(t)| dt \neq 0$  i  $\int_{I_2} |F(t)| dt \neq 0$ .
30 Wtedy istnieje rozwiązanie równania (i) z~dwoma wartościami zerowymi. \par

```

UWAGI

KOMENTARZ

Znaki $\$$ określają, które z komend działają wyłącznie w trybie matematycznym.

1: \ddot{x} jest matematycznym akcentem: $\$ \ddot{x}$ składa \ddot{x} .

5: $\!|$ składa ujemny wąski odstęp: nieznacznie dosuwa x do znaku różniczki. \leqno umieszcza z lewej strony równania jego numer. Obie komendy działają wyłącznie w trybie matematycznym. \rm składa antykwą numer równania.

13: Komenda \dot{x} jest kolejnym akcentem matematycznym: $\$ \dot{x}$ składa \dot{x} .

14: \equiv składa \equiv .

19: \rightarrow składa \rightarrow , ∞ składa ∞ .

23: \proclaim jest przykładem komendy wysokiego poziomu (makra \TeX -owego), w jej definicję włączono inne proste komendy. Widząc taką komendę \TeX czyta najpierw tekst do najbliższej kropki i składa go czcionką pogrubioną (zakłada się, że jest to tytuł twierdzenia). Pozostała część, do końca akapitu, będzie zło-

żona krojem pochyłym. Definicje, takie jak ta, pozwalają na rozkoszowanie się wszystkimi zaletami mocy programu, bez zagracania pliku dużą liczbą jawnych komend składu. *Taki powinien być ostateczny cel użytkownika programu \TeX .*

24: Komenda \rm jest wstawiona w tym miejscu w celu wymuszenia kroju antykwy w etykietach poszczególnych jednostek wyliczanych \item . \neq składa znak \neq .

26: \sup składa „sup”.

27: \inf składa „inf”, a \geq składa \geq .

28: \int składa znak całki \int . Komenda \limits umieszcza granice całkowania nad i pod znakiem całki. W niektórych operatorach granice są od razu umieszczone bezpośrednio nad i pod znakiem, wtedy komenda \nolimits ustawia je obok krawędzi górnej i dolnej. Patrz dodatek pod hasłem *matematyka: operacje*.

Pozostałe komendy w tej linii, $\!|$ i $\$, \$$, są komendami odstępów, a \forall składa \forall .

VII. ZEROWE ROZWIĄZANIA RÓWNIANIA $\ddot{x} + F(t)x = 0$

Pochodne i całki; justowanie równań.

Równanie

$$(i) \quad \frac{d^2x}{dt^2} + F(t)x = 0$$

opisuje zachowanie się wielu różnych układów. Dla $F = \text{const} > 0$ opisuje ono zwykły oscylator harmoniczny. Opisuje także ruch cząstki naładowanej w osiowo symetrycznym polu elektrycznym. (Tutaj F nie może być stałą ani nawet liczbą dodatnią¹³.) W obu przypadkach interesują nas rozwiązania oscylatora, tzn. rozwiązania, w których x przebiega wielokrotnie przez 0. Innym równaniem tego typu jest równanie ogniskowania geodetyki dla przyczynowych geodetyk w czasoprzestrzeni. Można dojść do niego przez wprowadzenie nowej zmiennej $y = (\dot{x}/x)$ ¹⁴. Wtedy

$$(ii) \quad \dot{y} \equiv \frac{dy}{dt} = -y^2 - F(t).$$

To równanie ma formę równania ogniskowania geodetyki^{14,15}. Interesujące nas w tym momencie rozwiązania (ze względu na ilość zastosowań) otrzymujemy, gdy istnieją t_1 i t_2 , gdzie $t_1 < t_2$, takie że $y \rightarrow \infty$, gdy $t \rightarrow t_1^+$, i $y \rightarrow -\infty$, gdy $t \rightarrow t_2^-$ (lub, równolegle, rozwiązania równania (i) z dwoma zerami). Następujące wyniki ustalają warunek na istnienie takiego rozwiązania¹⁵.

Twierdzenie. Niech $F(t)$ będzie funkcją ciągłą. Załóżmy, że

- a) istnieje dowolne t_0 , takie że $F(t_0) \neq 0$,
- b) istnieje dowolne $b > 0$, takie że dla dowolnych $t_1 < t_0$ i $t_2 > t_0$ istnieją przedziały domknięte $I_1 < t_1$ (tj. $\sup I_1 < t_1$) i $I_2 > t_2$ (tj. $\inf I_2 > t_2$) o długości $\geq b$, gdzie

$$\int_t^{t_0} F(t') dt' \geq 0 \quad \forall t \in I_1$$

oraz

$$\int_{t_0}^t F(t') dt' \geq 0 \quad \forall t \in I_2.$$

```

1 \noindent{\bf Dowód:} Zajmiemy się równaniem (ii) zamiast bezpośrednio (i).
2 Ponieważ  $F(t_0) \neq 0$ , istnieje zawierający  $t_0$  przedział  $[t_1, t_2]$ , taki
3 że  $F(t) \neq 0$  w przedziale  $[t_1, t_2]$ . Rozważmy rozwiązanie równania (ii)
4 dla  $y(t_0) = 0$ . Otrzymujemy
5 
$$\Delta = \min \left\{ \int_{t_1}^{t_0} y^2 dt, \int_{t_0}^{t_2} y^2 dt \right\} > 0$$

6 dla tego rozwiązania. \hfil\break
7 \indent Najpierw pokażemy, że  $y(t) \rightarrow -\infty$  dla dowolnego  $t > t_0$ .
8 W identyczny sposób można udowodnić, że  $y(t) \rightarrow \infty$  dla dowolnego
9  $t < t_0$ . \hfil\break
10 \indent Z warunku (b) wiemy, że istnieje ciąg przedziałów  $I_i$  o długości
11  $\geq b$ , gdzie  $\inf I_1 > t_2$  i  $\inf I_{i+1} > \sup I_i$ ,  $\forall i$ ,
12 taki że
13 
$$\int_{t_0}^{t_1} F(t) dt \geq 0 \quad \forall t \in I_i$$

14 Równanie (ii) można formalnie scałkować do granicy  $y(t_0) = 0$ ,
15 
$$\int_{t_0}^{t_2} y^2 dt - \int_{t_0}^{t_1} F(t) dt \leq \int_{t_2}^{t_1} y^2 dt - \int_{t_0}^{t_1} F(t) dt - \Delta$$

16 \noalign{\bigskip}
17 
$$\int_{t_2}^{t_1} y^2 dt - \frac{\Delta}{2} \geq 0 \quad \forall t \in I_i$$

18 
$$\int_{t_2}^{t_1} y^2 dt - \frac{\Delta}{2} < 0$$

19 \noalign{\smallskip}
20 \hbox{Zdefiniujmy}
21 \smallskip}
22 \alpha(t) = \int_{t_2}^{t_1} y^2 dt - \frac{\Delta}{2} < 0
23 \noalign{\smallskip}
24 \hbox{Wówczas}
25 \smallskip}
26 
$$\alpha(t) = -y^2 < -\frac{\Delta}{2} \quad \forall t \in I_i$$

27 
$$\alpha^2, \quad \forall t \in I_i$$

28

```

UWAGI

KOMENTARZ

Znaki $\$$ określają, które z komend działają wyłącznie w trybie matematycznym.

2: $\$ne$ składa \neq , a $\$, \$$ to mały odstęp.
5: $\$delta$ składa symbol δ , $\$min$ składa operator „min”. W tym wierszu widać, jakie są normalne pozycje granic całkowania.
5, 27: $\$Bigg$ składa duży ogranicznik. W tym przypadku ogranicznikami są $\{, \}, [i]$. Istnieje kilka komend typu przedrostkowego, np. $\$big, \$Big, \$bigg$ i $\$Bigg$; składają one ograniczniki coraz większego rozmiaru. Wszystkie są komendami działającymi w trybie matematycznym. (Duże ograniczniki powstają także za pomocą komend przedrostkowych $\$left$ i $\$right$, jak to zostało omówione w dodatku, pod hasłem matematyka: ograniczniki.)

10: $\$indent$ wymusza wcięcie akapitu w sytuacji, gdy nie powinien być on normalnie wcięty (np. wcześniejszy wiersz jest zakończony komendą $\$hfil\$break$, czyli TeX nadal jest w tym samym akapicie).

15–28: $\$leqalignno$ działa podobnie do komendy $\$eqalignno$, ale umieszcza numery równań po lewej stronie. Jak we wszystkich wyrównaniach, wpisy są wyrównywane w odniesieniu do znaków $\&$, a wiersze kończą komendy $\$cr$.

17, 20–22, itd.: $\$noalign$ pozwala tymczasowo zawiesić skład tabelaryczny; $\$hbox$ jest wykorzystany, aby nie składać danego fragmentu kursywą matematyczną.

18: $\$forall$ składa \forall , $\$, \$$ to komenda odstępu, a $\$in$ składa \in .

27: $\$alpha$ składa symbol α .

Wtedy istnieje rozwiązanie równania (i) z dwoma wartościami zerowymi.

Dowód: Zajmiemy się równaniem (ii) zamiast bezpośrednio (i). Ponieważ $F(t_0) \neq 0$, istnieje zawierający t_0 przedział $[t_1, t_2]$, taki że $F(t) \neq 0$ w przedziale $[t_1, t_2]$. Rozważmy rozwiązanie równania (ii) dla $y(t_0) = 0$. Otrzymujemy

$$\delta = \min \left\{ \int_{t_1}^{t_0} y^2 dt, \int_{t_0}^{t_2} y^2 dt \right\} > 0$$

dla tego rozwiązania.

Najpierw pokażemy, że $y(t) \rightarrow -\infty$ dla dowolnego $t > t_0$. W identyczny sposób można udowodnić, że $y(t) \rightarrow \infty$ dla dowolnego $t < t_0$.

Z warunku (b) wiemy, że istnieje ciąg przedziałów I_i o długości $\geq b$, gdzie $\inf I_1 > t_2$ i $\inf I_{i+1} > \sup I_i, \forall i$, taki że

$$\int_{t_0}^t F(t') dt' \geq 0 \quad \forall t \in I_i.$$

Równanie (ii) można formalnie scałkować do granicy $y(t_0) = 0$,

$$\begin{aligned} y(t) &= - \int_{t_0}^t y^2 dt' - \int_{t_0}^t F(t') dt' \leq - \int_{t_2}^t y^2 dt' - \int_{t_0}^t F(t') dt' - \delta \\ \text{(iii)} \quad &< - \int_{t_2}^t y^2 dt' - \frac{\delta}{2} \quad \forall t \in I_i. \end{aligned}$$

Zdefiniujmy

$$\alpha(t) = - \int_{t_2}^t y^2 dt' - \frac{\delta}{2} < 0.$$

Wówczas

$$\dot{\alpha}(t) = -y^2 < - \left[- \int_{t_2}^t y^2 dt' - \frac{\delta}{2} \right]^2 = -\alpha^2, \quad t \in I_i.$$

Dla każdego z przedziałów I_i , z dolną granicą a_i , można to równanie scałkować do postaci

$$-\alpha^{-1}(t) + \alpha^{-1}(a_i) < -(t - a_i).$$

Wtedy

$$\alpha(t) < [\alpha^{-1}(a_i) + (t - a_i)]^{-1}.$$

```

1 Dla każdego z przedziałów  $I_i$ , z dolną granicą  $a_i$ , można to
2 równanie scałkować do postaci
3 
$$\frac{d}{dt} \left( \alpha^{-1}(t) \alpha^{-1}(a_i) e^{-(t-a_i)} \right) =$$

4 
$$\alpha^{-1}(t) \alpha^{-1}(a_i) e^{-(t-a_i)}.$$

5 Teraz  $\alpha$  jest nierosnącą funkcją  $t$ . Poza tym w każdym  $I_i$ ,
6  $y(t) < -(\delta/2)$ . Więc  $\int_{I_i} y^2 dt < -(\delta^2 b/4)$ . Jeśli mamy  $n$ 
7 przedziałów  $I_i$  w przedziale  $(t_2, t)$ , będziemy mieć
8  $\alpha(t) < -n(\delta^2/4)$ . Czyli  $\alpha$  nie posiada granicy dolnej.
9 \hfil\break
10 \indent Wybierzmy dowolny przedział  $I_i$ , taki że  $\alpha(a_i) < -b^{-1}$ .
11 W tym przedziale  $\alpha^{-1}(a_i) + (t - a_i)$  będzie rozciągać się od
12  $\alpha^{-1}(a_i) < 0$  do  $\alpha^{-1}(a_i) + b > 0$ . Więc musi istnieć jakieś
13  $\bar{t}$ , takie że  $\alpha \rightarrow -\infty$ , gdy  $t \rightarrow \bar{t}$ .
14 Czyli  $y$  nie może być skończone wszędzie; zbadanie równania (ii) pokaże,
15 że musi ono dążyć do  $-\infty$ . \hfil\break
16 \indent Drugą część dowodu otrzymuje się w taki sam sposób. \clubsuit
17 \vfil\ejct

```

Teraz α jest nierosnącą funkcją t . Poza tym w każdym I_i , $y(t) < -(\delta/2)$.
Więc $-\int_{I_i} y^2 dt < -(\delta^2 b/4)$. Jeśli mamy n przedziałów I_i w przedziale (t_2, t) ,
będziemy mieć $\alpha(t) < -(nb\delta^2/4)$. Czyli α nie posiada granicy dolnej.

Wyberzmy dowolny przedział I_i , taki że $\alpha(a_i) < -b^{-1}$. W tym przedziale $\alpha^{-1}(a_i) + (t - a_i)$ będzie rozciągać się od $\alpha^{-1}(a_i) < 0$ do $\alpha^{-1}(a_i) + b > 0$.
Więc musi istnieć jakieś \bar{t} , takie że $\alpha \rightarrow -\infty$, gdy $t \rightarrow \bar{t}$. Czyli y nie może być skończone wszędzie; zbadanie równania (ii) pokaże, że musi ono dążyć do $-\infty$.

Drugą część dowodu otrzymuje się w taki sam sposób.♣


```

1 \centerline{\bf VIII. WIELE IMION DIRACA}
2 \centerline{\sl Wcięcia zawieszane.}
3 \headline{\hfil\it Przykład VIII\newpage}
4 \bigskip
5 {\baselineskip 18 true pt
6 \noindent Paul Adrien Maurice Dirac był jednym z~największych fizyków XX~wieku.
7 Oto lista kilku jego książek, zaczerpnięta z~\hbox{\it Books in Print,
8 1984--85\}: \bigskip
9 \hangindent=.5cm\hangafter=1
10 \noindent{\bf Dirac, P.}, {\it The Development of Quantum Theory.} J. Robert
11 Oppenheimer Memorial Prize Acceptance Speech. \dots
12 \vskip1pt
13 \hangindent=.5cm\hangafter=1
14 \noindent{\bf Dirac, P.A.}, {\it The Principles of Quantum Mechanics}. 4th ed.
15 (International Series of Monographs on Physics). \dots\hfil\break
16 --- {\it Spinors in Hilbert Space}. \dots
17 \vskip 1pt
18 \hangindent=.5cm\hangafter=1
19 \noindent{\bf Dirac, Pam, et al.}, {\it Directions in Physics: Lectures
20 Delivered During a Visit to Australia & New Zealand}. \dots
21 \vskip 1pt
22 \hangindent=.5cm\hangafter=1
23 \noindent{\bf Dirac, Paul A.}, {\it General Theory of Relativity}. \dots
24 \bigskip
25 \centerline{\$ \diamond \quad \diamond \quad \diamond \$}
26 \bigskip
27 \hangindent=-2,25 true in \hangafter=1
28 \noindent Odwiedzając Uniwersytet w~Wisconsin w~1929 roku, Dirac udzielał
29 wywiadu dla {\it Wisconsin State Journal}. Oto fragmenty:
30 \medskip{\sl
31 \hangindent=-2 true in \hangafter=-3
32 \noindent Słyszałem o~facecie, którego ma gościć Uniwersytet tej wiosny ---
33 fizyk matematyczny, ktoś, kto prześcignął Izaaka Newtona, Einsteina i~wszystkich
34 pozostałych. Pomyślałem, że dobrze zrobię, jeśli poproszę go o~wywiad dla
35 czytelników State Journal. Pan ten nazywa się Dirac i~jest Anglikiem.
36 \smallskip \hangindent=3 true in \hangafter=-4
37 \noindent ,,Profesorze'', zapytałem, ,,zauważyłem, że ma pan kilka liter przed
38 nazwiskiem. Czy reprezentują one coś szczególnego?'' ,,Nie'' odpowiedział
39 Dirac. ,,Chce pan powiedzieć, że mogę je dowolnie zinterpretować?''
40 ,,Tak'' --- odpowiedział. ,,Czy będzie w porządku, jeśli napiszę, że P.A.M.
41 oznacza Poincar'e Aloysius Mussolini?'' ,,Tak'' powiedział Dirac. ,,Świetnie'',
42 odpowiedziałem, ,,robimy bardzo szybkie postępy!'''.
43 \par}}\bye

```

UWAGI

Komendy „zawieszonych wcięć” w T_EX-u podlegają następującej konwencji: parametr `\hangindent` określa wielkość wcięcia. Dodatnia wartość wcinania akapitu z lewej strony, ujemna zaś z prawej. `\hangafter` określa, ile wierszy ma być wciętych. Załóżmy, że liczba wierszy oznaczonych do wcięcia wynosi n . Jeśli $n > 0$, wcięcie będzie następować *po* pierwszych n wierszach; jeśli $n < 0$, pierwsze n wierszy będzie wcięte. Komendy te muszą wyraźnie wystąpić w każdym akapicie, który ma być w ten sposób złożony. Niezależnie od działania tych komend, pierwszy wiersz jest automatycznie wcinany (w celu likwidacji tego wcięcia należy wykorzystać komendę `\noindent`).

5: Wartość parametru `\baselineskip` została zmieniła i będzie obowiązywać aż do linii 43 (do pojawienia się komendy `\par`).

9–23: Ten fragment przedstawia najczęściej spoty-

kane zastosowanie zawieszonych wcięć; `\hangafter` ma wartość 1, czyli wcinanie następuje od drugiego wiersza.

27: `\hangindent` jest ujemne (wcięcie będzie następować od prawego marginesu), a `\hangafter` dodatnie. Przykład tej strony składu niewiele różni się od zwykłego akapitu bez dodatkowych parametrów; w razie wątpliwości można poeksperymentować.

31, 36: `\hangafter` ma teraz wartość ujemną; można zobaczyć efekty działania tej komendy w połączeniu z dodatnią i ujemną wartością parametru `\hangindent`.

43: Komenda `\par` kończy akapit, zanim opuszczona będzie grupa, która ogranicza któreś z nowych ustawień. Komenda `\bye` jest preferowanym sposobem zakończenia pliku. Patrz uwagi do ostatniego wiersza na stronie xii.

VIII. WIELE IMION DIRACA

Wcięcia zawieszono.

Paul Adrien Maurice Dirac był jednym z największych fizyków XX wieku. Oto lista kilku jego książek, zaczerpnięta z *Books in Print, 1984–85*:

Dirac, P., *The Development of Quantum Theory*. J. Robert Oppenheimer Memorial Prize Acceptance Speech. ...

Dirac, P.A., *The Principles of Quantum Mechanics*. 4th ed. (International Series of Monographs on Physics). ...
— *Spinors in Hilbert Space*. ...

Dirac, Pam, et al., *Directions in Physics: Lectures Delivered During a Visit to Australia & New Zealand*. ...

Dirac, Paul A., *General Theory of Relativity*. ...

◇ ◇ ◇

Odwiedzając Uniwersytet w Wisconsin w 1929 roku, Dirac udzielał wywiadu dla *Wisconsin State Journal*. Oto fragmenty:

Słyszałem o facecie, którego ma gościć Uniwersytet tej wiosny — fizyk matematyczny, ktoś, kto prześcignął Izaaka Newtona, Einsteina i wszystkich pozostałych. Pomyślałem, że dobrze zrobię, jeśli poproszę go o wywiad dla czytelników *State Journal*. Pan ten nazywa się Dirac i jest Anglikiem.

„Profesorze”, zapytałem, „zauważyłem, że ma pan kilka liter przed nazwiskiem. Czy reprezentują one coś szczególnego?” „Nie” odpowiedział Dirac.

„Chce pan powiedzieć, że mogę je dowolnie zinterpretować?” „Tak” — odpowiedział. „Czy będzie w porządku, jeśli napiszę, że P.A.M. oznacza Poincaré Aloysius Mussolini?” „Tak” powiedział Dirac. „Świetnie”, odpowiedziałem, „robimy bardzo szybkie postępy!”.

```

1 \input tbeoutma \pageno=31          % W TYM MIEJSCU ROZPOCZYNA SIĘ NOWY PLIK
2 \headline{\hfil\it Przykład IX\newpageno}
3 %UWAGA ,, \newpageno'' nie jest standardową komendą TeX'a; patrz str. 2.
4
5 \centerline{\bf IX. Z LISTÓW RAMANUJANA DO HARDY'EGO}
6 \centerline{\sl Szczegóły składu równań.} \medskip
7 \noindent Na początku 1913 roku, Srinivasa Ramanujan, matematyczny samouk,
8 zaczął pisać listy do G.H. Hardy'ego w Cambridge, przedstawiając mu odkryte
9 przez siebie twierdzenia. Listy zawierają około 120 wyników, ogłoszonych bez
10 dowodu. Poniżej znajduje się kilka z nich, selekcji dokonał Hardy16$.
11 $$\displaylines{
12 (1)\quad 1 - \{3!\over(1!2!)^3\}x^2+ \{6!\over(2!4!)^3\}x^4 - \cdots \hfill \cr
13 \hfill = \biggl( \sum_{n=0}^{\infty} \{x^n\over(n!)^3\} \biggr)
14 \biggl(1-\{x\over(1!)^3\}+\{x^2\over(2!)^3\}-\cdots \biggr) \cr
15 \noalign{\vfil}
16 (2)\hfill 1+ \sum_{n=1}^{\infty} (-1)^n (1+4n) \left\{ \prod_{j=1}^n
17 \{2j-1\over 2j\} \right\}^{\!3} = \{2\over\pi\} \hfill \cr
18 \noalign{\vfil}
19 (3)\hfill 1+9\left( \{1\over 4\} \right)^4 + 17\left( \{1\cdot5\over4\cdot8\}
20 \right)^4+25\left( \{1\cdot5\cdot9\over4\cdot8\cdot12\} \right)^4 + \cdots =
21 \{2^{\!3\over2}\} \over \pi^{\!1/2} \{ \Gamma( \{3\over 4\} ) \}^{\!2} \hfill \cr
22 \noalign{\vfil}
23 (4)\hfill 1 - 5\bigg( \{1\over2\} \bigg)^{\!5} + 9\bigg( \{1\cdot3\over2\cdot4\}
24 \bigg)^{\!5} - 13\bigg( \{1\cdot3\cdot5\over2\cdot4\cdot6\} \bigg)^{\!5} +
25 \cdots = \{2 \over \{ \Gamma( \{3\over 4\} ) \} \}^{\!4} \hfill \cr
26 \noalign{\vfil}
27 (5)\hfill \int^{\infty}_{\scriptstyle 0} \{dx \over (1 + x^2)(1+r^2x^2)(1+r^4x^2)\dots} =
28 \{\pi \over 2\sum_{j=0}^{\infty} (r^0\cdots r^j)\} \hfill \cr
29 \noalign{\vfil}
30 (6)\quad \hbox{Jeśli}; \alpha \beta = \pi^2, \> \hbox{to} \hfill \cr
31 \hfil \alpha^{\!-\{1\over4\}} \Bigl\lgroup 1+4\alpha \int^{\infty}_0 \{x^{\!-\alpha}
32 x^2\} \over e^{\!2\pi x} -1\} dx \Bigl\rgroup = \beta^{\!-\{1\over4\}} \Bigl\lgroup 1+
33 4\beta \int^{\infty}_0 \{x^{\!-\beta} x^2\} \over e^{\!2\pi x} -1\} dx \Bigl\rgroup \cr
34 \noalign{\vfil}
35 (7)\, \{ \bf \int\limits^{\infty}_0 \{1+(\{x\over b+1\})^2 \over 1 + (\{x\over a\})^2\}
36 \cdot \{1 + (\{x\over b+2\})^2 \over 1 + (\{x\over a+1\})^2\} \cdots dx =
37 \{\sqrt{\pi\over2}\} \{ \Gamma(a+\{1\over2\}) \Gamma(b+1) \Gamma(b-a+\{1\over2\})
38 \over \Gamma(a) \Gamma(b+\{1\over2\}) \Gamma(b-a+1)\} \} \hfill \cr
39 \noalign{\eject}

```

UWAGI

JEŚLI DO TEJ PORY NIE SPOTKAŁEŚ SIĘ Z MATEMATYKĄ W T_EX-U, TA STRONA MOŻE WYDAĆ CI SIĘ ZBYT SKOMPLIKOWANA; PRZEJRZYJ NAJPIERW WCZEŚNIEJSZE PRZYKŁADY. W tym miejscu rozpoczyna się nowy plik. `\input tbeoutma` włącza takie same komendy formatujące jak dla str. 1. Dziwna początkowa wartość `\pageno` odnosi się do nowej numeracji stron, omówionej na str. 2.

11, itd.: `\displaylines` eksponuje równania, każde wyśrodkowane w osobnym wierszu. Nie musimy więc wstawiać znaków `$$` przed i po każdym równaniu. `\hfill` przesuwają równania do prawej lub do lewej strony; `\hfil` tu nie będzie działać. Nawiasy, klamry, itp. są poprzedzone komendami `\left`, `\right`, `\biggl`, itp. Komendy `\big`, `\Big`, `\bigg` i `\Bigg` zmieniają wielkość takich ograniczników, dają specyficzne rozmiary, z dodatkowym `l` lub `r` określają lewy lub prawy ogranicznik; komendy `\left` i `\right` dają ograniczniki, które „rosną” automatycznie z pionowym rozmiarem równania. Patrz matematyka: ograniczniki w dodatku.

13, 16: `$$\sum` składa znak \sum . Pozycja granic zależy od miejsca, gdzie pojawia się symbol: porównaj równa-

nia 1 i 5. W wierszu 16 `\nolimits` przesuwają granice z górnej pozycji (równanie 2). `$$\prod` składa \prod .

15, itd.: `\noalign` pozwala na umieszczenie materiału, który będzie ignorował wyrównywanie nakazane przez komendę `\displaylines`.

17, 23, 24: `$$\!` składa mały „ujemny” odstęp. W tym miejscu umożliwia właściwe umieszczenie pozycji wykładników.

25: `\vphantom` składa odstęp o zerowej szerokości, ale z wysokością równą w tym przypadku „`\big(`”; pozwala to T_EX-owi na umieszczenie wykładnika wyżej. Porównaj równania 3 i 4.

27: `\raise` i `\lower` przesuwają w górę i w dół zawartość pudełka. Standardowym krojem wewnątrz pudełka jest `\textstyle`. Porównaj granice całki.

32: `\hfil` nie jest wystarczająco mocne, by w tym miejscu wykonać justowanie do prawej: patrz równanie 6.

32–34: `\lgroup`, `\rgroup` składają nawiasy pogrubione.

36: `\bf` zmienia tylko niektóre znaki na pogrubione. `\limits` przemieszcza granice na krańce całki.

IX. Z LISTÓW RAMANUJANA DO HARDY'EGO

Szczegóły składu równań.

Na początku 1913 roku, Srinivasa Ramanujan, matematyczny samouk, zaczął pisać listy do G.H. Hardy'ego w Cambridge, przedstawiając mu odkryte przez siebie twierdzenia. Listy zawierają około 120 wyników, ogłoszonych bez dowodu. Poniżej znajduje się kilka z nich, selekcji dokonał Hardy¹⁶:

$$(1) \quad 1 - \frac{3!}{(1!2!)^3}x^2 + \frac{6!}{(2!4!)^3}x^4 - \dots = \left(\sum_{n=0}^{\infty} \frac{x^n}{(n!)^3} \right) \left(1 - \frac{x}{(1!)^3} + \frac{x^2}{(2!)^3} - \dots \right)$$

$$(2) \quad 1 + \sum_{n=1}^{\infty} (-1)^n (1+4n) \left\{ \prod_{j=1}^n \frac{2j-1}{2j} \right\}^3 = \frac{2}{\pi}$$

$$(3) \quad 1 + 9 \left(\frac{1}{4} \right)^4 + 17 \left(\frac{1 \cdot 5}{4 \cdot 8} \right)^4 + 25 \left(\frac{1 \cdot 5 \cdot 9}{4 \cdot 8 \cdot 12} \right)^4 + \dots = \frac{2^{\frac{3}{2}}}{\pi^{1/2} \{\Gamma(\frac{3}{4})\}^2}$$

$$(4) \quad 1 - 5 \left(\frac{1}{2} \right)^5 + 9 \left(\frac{1 \cdot 3}{2 \cdot 4} \right)^5 - 13 \left(\frac{1 \cdot 3 \cdot 5}{2 \cdot 4 \cdot 6} \right)^5 + \dots = \frac{2}{\{\Gamma(\frac{3}{4})\}^4}$$

$$(5) \quad \int_0^{\infty} \frac{dx}{(1+x^2)(1+r^2x^2)(1+r^4x^2)\dots} = \frac{\pi}{2 \sum_{j=0}^{\infty} (r^0 \dots r^j)}$$

(6) Jeśli $\alpha\beta = \pi^2$, to

$$\alpha^{-\frac{1}{4}} \left(1 + 4\alpha \int_0^{\infty} \frac{x e^{-\alpha x^2}}{e^{2\pi x} - 1} dx \right) = \beta^{-\frac{1}{4}} \left(1 + 4\beta \int_0^{\infty} \frac{x e^{-\beta x^2}}{e^{2\pi x} - 1} dx \right)$$

$$(7) \quad \int_0^{\infty} \frac{1 + (\frac{x}{b+1})^2}{1 + (\frac{x}{a})^2} \cdot \frac{1 + (\frac{x}{b+2})^2}{1 + (\frac{x}{a+1})^2} \dots dx = \frac{\sqrt{\pi} \Gamma(a + \frac{1}{2}) \Gamma(b+1) \Gamma(b-a + \frac{1}{2})}{2 \Gamma(a) \Gamma(b + \frac{1}{2}) \Gamma(b-a+1)}$$

```

1 (8)\hfill 4\int^{\infty}_0 {x^{-x\sqrt{5}} \over \cosh x} dx
2 \phantom{a} = \hphantom{a} {1 \over \displaystyle 1 + {1^2\hfill
3 \over \displaystyle 1 + {1^2 \over \textstyle 1 + {2^2 \over
4 \textstyle 1 + {2^2 \over 1 + {3^2 \over 1 + {3^2 \over 1 +
5 \cdots}}}}}} \hfill \cr \noalign{\vfil} (9)\hfill {\rm Jeśli}
6 \hfill\qqquad\cr \underbrace{\overbrace{u={x\over 1+} {x^5\over 1+}
7 {x^{10}\over 1+} {x^{15}\over 1+\cdots} \quad {\rm i} \quad v =
8 {x^{1\over 5}\over 1+} {x\over 1+} {x^2\over 1+} {x^3\over 1+\cdots}}},
9 \cr {\rm to}\cr v^5 = u{1-2u+4u^2-3u^3+u^4 \over
10 1+3u+4u^2+2u^3+u^4}\cr \noalign{\vfil} (10)\hfill {1\over 1+}
11 {e^{-2\pi}\over 1+} {e^{-4\pi}\over 1+\cdots} = \left\{
12 \sqrt{\smash{\Biggl( {5+\sqrt{5}\over 2} \Biggr)}} \vphantom{\bigg{}}-
13 {\sqrt{5+1}\over 2} \right\} e^{\{2\over 5\} \pi} \hfill \cr
14 \noalign{\vfil}
15 (11)\hfill {1\over 1+} {e^{-2\pi\sqrt{5}\over 1+} {e^{-4\pi\sqrt{5}\over 1+\cdots}
16 = \Biggl[ {\sqrt{5}\over 1+\sqrt{5}\of{\Big\{ 5^{3/4} \big( {\sqrt{5-1}\over 2} \big)
17 ^{5/2} - 1 \Big\}}} - {\sqrt{5+1}\over 2} \Bigg] e^{2\pi/\sqrt{5}} \hfill \cr
18 \noalign{\vfil}
19 (12)\;{\rm Jeśli}\;F(k)=1+\big( {\textstyle{1\over 2}} \big)^{2k} + \big(
20 {\textstyle{1\cdot 3\over 2\cdot 4}} \big)^{2 k^2} + \cdots\;{\rm i}\;
21 F(1-k)=\sqrt{210}F(k), \;{\rm to} \hfill \cr
22 \noalign{\smallskip}
23 \qqquad\qqquad k = (\sqrt{2-1})^4 (2-\sqrt{3})^2 (\sqrt{7}-\sqrt{6})^4 (8-3\sqrt{7})^2
24 \hfill\cr
25 \hfill\times (\sqrt{10}-3)^4 (4-\sqrt{15})^4 (\sqrt{15}-\sqrt{14})^2
26 (6-\sqrt{35})^2 \qqquad \cr
27 \noalign{\vfil}
28 (13)\;{\rm Ilość liczb pomiędzy}\;A\;{\rm i}\;x,\;{\rm które}
29 są albo kwadratami, albo sumą}\hfill \cr
30 \qqquad {\rm dwóch kwadratów, wynosi}\hfill \cr
31 K \int^x_A {dt\over \sqrt{\log t}} + \theta(x), \cr \noalign{\smallskip}
32 \qqquad{\rm gdzie}\;K=0,764 \ldots\;{\rm a}\;{\rm \theta}(x)\;{\rm jest}
33 nieskończenie mała.}\hfill\cr}$$
34 \vfil\ejct

```

UWAGI

KOMENTARZ

Tak jak i poprzednia strona, ta jest aż „ciężka” od matematyki. Jeśli to Twoje pierwsze kroki w \TeX -u, odwróć ją i udawaj, że niczego nie widziałeś. Znaki $\$$ w uwagach na tej i poprzedniej stronie przypominają, że dana komenda działa wyłącznie w trybie matematycznym.

2–5: `` składa odstęp o wysokości, szerokości i odsadce równej **a**. `\hphantom` działa podobnie, ale składa odstęp o zerowej wysokości. Porównaj z `\vphantom` z wiersza 25 na poprzedniej stronie. Rozmiar licznika i mianownika po prawej stronie znaku równości w równaniu 8 może być regulowany przez zastosowanie komend `\displaystyle` lub `\textstyle`. Domyślnie i licznik, i mianownik są bardzo małe (`\scriptscriptstyle`). Normalne wyśrodkowanie liczników i mianowników można osiągnąć za pomocą komendy `\hfill`. Jeśli odstęp pionowy wydaje się być za mały (jak tutaj), należy umieścić dodatkowy odstęp komendą `\vphantom`.

5–9: Antykwę uzyskuje się komendą `\rm`. Umieszczone w wierszu 7 `\qqquad` równoważy odległości zaj-

mowane przez „(9)” z lewej strony wiersza na stronie wynikowej; w rezultacie słowo „Jeśli” jest z grubsza wyśrodkowane.

12: `\smash` daje odwrotny efekt niż `\vphantom`: składa element o zerowej wysokości. W połączeniu z komendą `\vphantom` może być wykorzystane do pełnej kontroli odstępów pionowych. W tym miejscu naturalna wysokość wzoru jest ustawiona na zero komendą `\smash`, a wysokość, równa `\bigg(`, przypisana jest komendą `\vphantom`. W wyniku połączenia tych komend uzyskuje się okropny odstęp pionowy, ale tylko w taki sposób można pokazać efekt działania kombinacji `\smash-\vphantom`.

16–18: Prawa strona równania 11 mogłaby wyglądać korzystniej po dopasowaniu rozmiaru nawiasów kwadratowych.

28–33: Dłuższe partie tekstu najlepiej jest umieszczać wewnątrz `\hbox`. Przywraca to standardowe dla tekstu odstępy (`\rm` zmienia krój pisma, ale zachowuje matematyczny tryb odstępów, tak więc wszystkie spacje są likwidowane). Podziału wiersza wewnątrz `\hbox` można dokonać tylko ręcznie.

$$(8) \quad 4 \int_0^\infty \frac{x e^{-x\sqrt{5}}}{\cosh x} dx = \frac{1}{1 + \frac{1^2}{1 + \frac{1^2}{1 + \frac{2^2}{1 + \frac{2^2}{1 + \frac{3^2}{1 + \frac{3^2}{1 + \dots}}}}}}}$$

$$(9) \quad \text{Jeśli}$$

$$\underbrace{u = \frac{x}{1+} \frac{x^5}{1+} \frac{x^{10}}{1+} \frac{x^{15}}{1+\dots}} \quad \text{i} \quad \underbrace{v = \frac{x^{\frac{1}{5}}}{1+} \frac{x}{1+} \frac{x^2}{1+} \frac{x^3}{1+\dots}},$$

to

$$v^5 = u \frac{1 - 2u + 4u^2 - 3u^3 + u^4}{1 + 3u + 4u^2 + 2u^3 + u^4}$$

$$(10) \quad \frac{1}{1+} \frac{e^{-2\pi}}{1+} \frac{e^{-4\pi}}{1+\dots} = \left\{ \sqrt{\left(\frac{5 + \sqrt{5}}{2} \right)} - \frac{\sqrt{5} + 1}{2} \right\} e^{\frac{2}{5}\pi}$$

$$(11) \quad \frac{1}{1+} \frac{e^{-2\pi\sqrt{5}}}{1+} \frac{e^{-4\pi\sqrt{5}}}{1+\dots} = \left[\frac{\sqrt{5}}{1 + \sqrt[5]{\left\{ 5^{3/4} \left(\frac{\sqrt{5}-1}{2} \right)^{5/2} - 1 \right\}}} - \frac{\sqrt{5} + 1}{2} \right] e^{2\pi/\sqrt{5}}$$

(12) Jeśli $F(k) = 1 + \left(\frac{1}{2}\right)^2 k + \left(\frac{1\cdot 3}{2\cdot 4}\right)^2 k^2 + \dots$ i $F(1-k) = \sqrt{210}F(k)$, to

$$k = (\sqrt{2} - 1)^4 (2 - \sqrt{3})^2 (\sqrt{7} - \sqrt{6})^4 (8 - 3\sqrt{7})^2 \\ \times (\sqrt{10} - 3)^4 (4 - \sqrt{15})^4 (\sqrt{15} - \sqrt{14})^2 (6 - \sqrt{35})^2$$

(13) Ilość liczb pomiędzy A i x , które są albo kwadratami, albo sumą

dwóch kwadratów, wynosi

$$K \int_A^x \frac{dt}{\sqrt{(\log t)}} + \theta(x),$$

gdzie $K = 0,764\dots$, a $\theta(x)$ jest nieskończenie mała.

```

1 \centerline{\bf X. TYPOGRAFIA}
2 \centerline{\sl Odmiany i~stopnie krojów pisma.}
3 \headline{\hfil\it Przykład X\newpageno}
4 \bigskip
5 \font\sans=plss10
6 \font\sansbf=plssbx10
7 \font\nice=pldunh10
8 \font\lrm=plr7 scaled\magstep1
9 \font\bbf=plbx5 scaled\magstep3
10
11 \nice
12 \noindent Typografia jest sztuką konserwatywną: styl nazywany w~angielskiej
13 terminologii typograficznej „modern” (nowoczesny) ma 200 lat. Pierwsze
14 czcionki tego stylu powstały u~schyłku 18. wieku, blisko końca okresu
15 oświecenia i~początku rewolucji przemysłowej. Głównymi inicjatorami stylu byli
16 Giambattista Bodoni, drukarz i~stemplarz z~Parmy, oraz Firmin Didot, odlewnik
17 czcionek, członek sławnej paryskiej rodziny drukarzy. Byli to współcześni
18 Thomasa Jeffersona, tak więc jeśli dzisiaj uważamy Jeffersona za człowieka
19 nowoczesnego pod wieloma względami, może nie będzie przesadą nazwanie również
20 czcionek z~jego czasów nowoczesnymi (modern).
21 \medskip \lrm
22 \noindent{\bbf -- Charles A. Bigelow,}\hfil\break
23 wprowadzenie do książki {\sans Computer Modern Typefaces}
24 {\sansbf Donalda Knutha,}\hfil\break
25 \rm \copyright\ 1986 Addison-Wesley Publishing Company, Inc. \hfil\break
26 Przedrukowano za zgodą wydawcy.
27 \vfil\ejct

```

UWAGI

O FONTACH

Przykład ten pokaże, jak uzyskać dostęp do krojów, które nie są wstępnie załadowane w \TeX -u. Ponieważ terminologia nie zawsze jest używana we właściwy sposób, przypomnijmy za *Leksykonem pism drukarskich* A. Tomaszewskiego (1996), że *czcionka* to rodzaj nośnika pisma drukarskiego, metalowy prostopadłościan odlany ze stopu drukarskiego, *font* to zapisany w postaci cyfrowej obraz kroju pisma, czyli zestaw informacji o kształtach poszczególnych liter i znaków danego kroju, a *krój pisma* to charakterystyczny obraz pisma drukarskiego o jednolitych cechach graficznych (przyp. tłum.). Wielkość pisma (stopień) podawana jest zwykle w punktach. Na przykład 10-punktowy *Times-Roman* jest przykładem prostego pisma (antykwy), o rozmiarze 10 punktów, wykorzystywanego przez gazetę London Times. Dany krój pisma może wystąpić w kilku odmianach: *pogrubionej*, *kursywie* czy *pochylej*. W \TeX -u automatycznie ładowane są standardowe fonty: antykwa (z odmianami pogrubioną, kursywą i pochylą), pismo maszynowe, i kilka zestawów symboli matematycznych. Wszystkie te kroje należą do rodziny Computer Modern, stworzonej głównie przez Donalda Knutha. Przykład, w jaki sposób można uzyskać dodatkowe fonty Computer Modern, omówiono poniżej.

5–9: W wierszach tych znajdują się definicje wykorzystywanych krojów pisma. Są one utworzone za pomocą komendy `\font`, która po prawej stronie znaku = zawiera *zewnętrzną* nazwę fontu (tzn. nazwę, pod którą w osobnym pliku znajduje się informacja dotycząca fontu) i nazwę, którą wykorzystamy w dokumencie. Informacja o foncie jest zwykle zawarta

w dwóch plikach: pierwszy plik, *metryczny*, zawiera informację dotyczącą rozmiaru każdego ze znaków; drugi — informację o ich kształcie, odpowiednią do poziomu rozdzielczości konkretnego urządzenia podglądu czy drukującego, które wykorzystujemy. Jeśli zastosujemy nazwę zewnętrzną, dla której brak takiego pliku, w momencie przetwarzania tekstu otrzymamy wiadomość o błędzie. Jeśli skorzystamy z fontów, których drukarka nie potrafi wyrysować, wiadomość o błędzie otrzymamy w momencie próby drukowania dokumentu.

5: `\sans` to definicja bezszeryfowego kroju pisma, 10pt Computer Modern.

6, 7: Podobnie jak w linii 5, zdefiniowano font 10pt Computer Modern półgruby rozszerzony bezszeryfowy oraz font 10pt Computer Modern „dunhill” (styl popularyzowany przez papierosy o tej nazwie).

8–9: Komendy te ładują, odpowiednio, 7pt antykwę i 5pt półgruby rozszerzony, każdy powiększony przez współczynnik skali. `\magstep1` powiększa 1,2 raza a `\magstep3` — $1,2^3$ (= 1,728). Można także jawnie określić stopień pisma, tak jak to zrobiono na str. viii.

11: Tekst składany jest krojem o nazwie `\nice`. Ponieważ nie korzystamy z żadnych symboli grupujących (np. `{ i }`), taki krój będzie obowiązywać, dopóki nie zostanie jawnie zmieniony.

21: Zmiana bieżącego kroju `\nice` na `\lrm`.

22–25: Zmiana kroju określonych fragmentów tekstu na `\bbf`, `\sans` i `\sansbf` i wreszcie na krój standardowy `\rm` (10pt antykwa).

WYKAZ FONTÓW RODZINY COMPUTER MODERN ZNAJDUJE SIĘ W DODATKU POD HASŁEM fonty.

X. TYPOGRAFIA

Odmiany i stopnie krojów pisma.

Typografia jest sztuką konserwatywną: styl nazywany w angielskiej terminologii typograficznej „modern” (nowoczesny) ma 200 lat. Pierwsze czcionki tego stylu powstały u schyłku 18. wieku, blisko końca okresu oświecenia i początku rewolucji przemysłowej. Głównymi inicjatorami stylu byli Giambattista Bodoni, drukarz i stemplarz z Parmy, oraz Firmin Didot, odlewnik czcionek, członek sławnej paryskiej rodziny drukarzy. Byli to współcześni Thomasa Jeffersona, tak więc jeśli dzisiaj uważamy Jeffersona za człowieka nowoczesnego pod wieloma względami, może nie będzie przesadą nazwanie również czcionek z jego czasów nowoczesnymi (modern).

– **Charles A. Bigelow,**

wprowadzenie do książki *Computer Modern Typefaces* **Donalda Knutha,**

© 1986 Addison-Wesley Publishing Company, Inc.

Przedrukowano za zgodą wydawcy.


```

1 \centerline{\bf XI. LIST}
2 \centerline{\sl Definicje; listy.}
3 \headline{\hfil\it Przykład XI\newpageno}%\newpageno KOMENDA WŁASNA
4 \bigskip
5
6 \def\letterhead #1{\hrule height 1pt
7     \medskip
8     \vbox{\baselineskip=12pt\halign{##\hfil\cr #1}}}
9
10 \def\date #1{\vbox{\hsize 1,5true in\noindent #1}\bigskip}
11
12 \def\address{\rightline
13     {\vbox{\hsize 1,5true in\baselineskip=12pt \parindent=0pt
14     Nazwisko adresata\hfil\break Ulica \hfil\break Miasto \hfil\break
15     Kod pocztowy \hfil\break Kraj}}}
16
17 \def\salm #1{\noindent #1}
18
19 \def\endletter #1#2{\nobreak\medskip\rightline{\vbox{\hsize 1,5true in
20     \parindent=0pt #1\vskip 1,25 true cm #2}}}
21
22 \letterhead{Nazwisko nadawcy\cr Adres nadawcy \cr Itd.\cr}
23 \date{Data}
24 \address
25
26 \salm{Drogi Czytelniku}
27
28 To jest list bez zawartości. Przykład pisania listów, który umieściłem
29 w poprzedniej wersji niniejszej książki, miał dużo bogatszą zawartość,
30 ale mądrzy ludzie powiedzieli mi, że treść odwraca uwagę czytelnika
31 od formy.
32
33 {\sl Forma\}/} w tym przypadku to coś bardzo ważnego.
34 \endletter{Pozdrowienia,}{Autor.}
35 \vfil\eject

```

UWAGI

6–20: W wierszach tych pokazano, jak można definiować własne komendy za pomocą składni: `\def`, nazwa nowej komendy (włączając możliwe parametry), a dalej „tekst do zastąpienia”, ujęty w klamry. Jest to albo tekst, albo zestaw instrukcji, które określają nową komendę. Prosty przykład nowej komendy, która otwiera tekst często powtarzający się. Tak więc jeśli masz nawyk pisania długich i serdecznych, ale podobnych listów do różnych ludzi, możesz zdefiniować

```
\def\passion {długi tekst}
```

i wtedy pisać tylko `\passion` za każdym razem.

6–15: Komendy z tych wierszy składają nagłówek listu: program składa poziomą kreskę, i umieszcza z lewej strony pudełko (`\vbox`) o rozmiarze poziomym 1,5 cala, zawierające wiersze dotyczące danych nadawcy: nazwisko, adres, itd. Odstęp między wierszami wewnątrz pudełka został zmieniony (za pomocą komendy `\baselineskip`) na 12 punktów, a wcięcie akapitu (`\parindent`) ustalono na 0 punktów. Zmiany takie działają tylko wewnątrz pudełka `\vbox` lub `\hbox`. Pudełka są omówione w przykładzie XV oraz w dodatku.

10: Podobnie jak `\letterhead`, tworzone jest pudełko `\date` o rozmiarze 1,5 cala, wyrównane do lewego marginesu. `\noindent` likwiduje wcięcie akapitu. Ta komenda ma parametr oznaczony przez `#1`,

który należy określić przy jej wywołaniu. Patrz linia 23.

12: `\address` wykonuje skład tabeli za pomocą komendy `\halign`. Automatycznie wyrównuje materiał, na którym działa. Zamiast zwykłego znaku `#` we wstępie, do komendy `\halign` muszą być w tym miejscu użyte `##` (patrz jak komenda `\halign` jest wykorzystana w przykładzie VI).

17: `\salm` to komenda składająca pozdrowienia.

19, 20: W wierszach tych znajduje się definicja dwuparametrowej komendy. Przy definiowaniu komend można użyć nie więcej niż 9 parametrów. `\nobreak` nie pozwala na złamanie strony w miejscu `\medskip`.

22–34: Ten fragment przedstawia sposób wykorzystania komend zdefiniowanych wyżej. Definicje takich komend zwykle umieszczone są w osobnym pliku, który może być wykorzystany jako *wsad* (*input*). Autor książki ma zbiór komend przydatnych przy pisaniu listów, zawarty w pliku *letter.tex*. Po umieszczeniu na początku pliku, w którym będzie pisany list, zwrotu `\input letter`, może wywoływać zdefiniowane w nim komendy `\letterhead`, `\address`, itd. **23:** Datę w tym miejscu należy wpisać ręcznie. W dodatku, pod hasłem `\year` znajduje się komenda umieszczająca automatycznie datę systemową.

XI. LIST*Definicje; listy.*

Nazwisko nadawcy

Adres nadawcy

Itd.

Data

Nazwisko adresata

Ulica

Miasto

Kod pocztowy

Kraj

Drogi Czytelniku

To jest list bez zawartości. Przykład pisania listów, który umieściłem w poprzedniej wersji niniejszej książki, miał dużo bogatszą zawartość, ale mądrzy ludzie powiedzieli mi, że treść odwraca uwagę czytelnika od formy.

Forma w tym przypadku to coś bardzo ważnego.

Pozdrowienia,

Autor.

```

1 \centerline{\bf XII. NOTKA}
2 \centerline{\sl Więcej definicji; przechowywanie zmiennych; komendy pętli;
3 notki.}
4 \headline{\hfil\it Przykład XII\newpageno}%\newpageno --- KOMENDA WŁASNA
5 \bigskip
6
7 \newcount\thusfar
8 \def\pattern #1#2{\thusfar=0
9   \loop #1 \advance\thusfar by 1 \ifnum\thusfar<#2 \repeat}
10
11 \def\block #1{\vbox{\hsize 2,5 true cm\noindent \bf#1:}}
12
13 \def\memo DLA:#1OD:#2TEMAT:#3DATA:#4\par{\centerline{\sl PAMIĘTAJ!}
14   \bigskip \hrule height1pt \medskip
15   \vbox{\parindent=75pt\parskip=1pt
16     \item{\block{DLA}}#1
17     \item{\block{OD}}#2
18     \item{\block{TEMAT}}#3
19     \item{\block{DATA}}#4\medskip \hrule height1pt \bigskip}
20
21 \memo
22 DLA: Dla Ciebie.
23 OD: Ode mnie.
24 TEMAT: Pisanie eleganckich notek.
25 DATA: Dzisiaj.
26
27 Oto jeden ze sposobów tworzenia notki. Główną jej część wpisuje się tak jak
28 każdy inny tekst w \TeX-u: pusty wiersz oznacza nowy akapit,
29 {\tt\string\noindent} likwiduje wcięciu akapitu, itd.
30
31 Mam nadzieję, że jest to pouczająca notka.
32 \bigskip
33 \centerline{\pattern{${\star$}}{25}}
34 \vfil\eject

```

UWAGI

7: Użytkownik ma możliwość umieszczenia pewnych informacji w wewnętrznych rejestrach \TeX -a. Rejestr jest klasyfikowany według tego, czy zawiera zwykłe liczby, wielkości, które mają wymiar (np. rozmiar, wielkość), czy też całe pudełka (\vbox i \hbox), itd. Każdy rejestr w danej klasie ma etykietę w postaci liczby. Rejestry zawierające zwykłe liczby nazwane są \count i mają etykiety $\count0$, $\count1$, itd. Ponieważ niektóre z rejestrów są już wykorzystane przez program do umieszczenia np. numerów stron ($\count0$), \TeX za pomocą komendy \newcount przydziela nowy, nie wykorzystany wcześniej rejestr, kiedy chcemy coś zapamiętać. \thusfar reprezentuje tu liczbę będącą etykietą nowego rejestru, ale nie potrzebujemy znać tej liczby, ani nawet wiedzieć co to takiego: wykorzystujemy nazwę \thusfar w miejsce liczby. Więcej informacji dotyczących przydzielania rejestrów znajduje się w dodatku, pod hasłem \newcount .

8–9: \pattern składa $\#2$ kopii symbolu czy tekstu wyspecyfikowanego parametrem $\#1$. Ustawia zero dla zmiennej w rejestrze \thusfar , a następnie wykonuje komendy $\loop\dots\repeat$ i składa kopie $\#1$. Po złożeniu każdej kolejnej kopii, zmienna oznaczona \thusfar wzrasta (dzięki \advance) o 1 i cały pro-

ces powtarza się aż do momentu, kiedy zmienna ta nie osiągnie wartości $\#2$. \ifnum jest jedną z kilku komend warunkowych w \TeX -u: porównuje liczby.

11: \block składa pudełko o stałym rozmiarze poziomym, by wykorzystać je w \memo .

13–19: W wierszach tych zdefiniowana jest kombinacja komend: \memo DLA: \dots \par . Mogą one być użyte wprost, bez dodatkowych klamer. \memo powoduje, że \TeX szuka słów „DLA:”, „OD:”, itd. i umieszcza tekst, który po nich następuje we właściwym miejscu. Działanie komendy kończy się wraz z końcem akapitu. Wszystko musi być podane dokładnie, tak jak w liście komend: wersaliki, dwukropki. Takie komendy „wysokiego poziomu” mogą być bardzo przydatne do składu dokumentów za pomocą kilku komend (wszystkie uciążliwe detale formatowania ukryte są wewnątrz). To, co jest w rzeczywistości drukowane, jest tekstem parametru $\#1$ komendy \block . Nie jest konieczne, by słowa w \block były takie same jak w nazwie komendy.

21–25: Przykład działania komendy \memo . Słowa DLA:, itd. muszą być wpisane dokładnie tak jak tutaj, ale tekst po każdym dwukropku może być dowolny. Tekst notki rozpoczyna się po pustym wierszu.

33: Sposób wykorzystania komendy \pattern .

XII. NOTKA

Więcej definicji; przechowywanie zmiennych; komendy pętli; notki.

PAMIĘTAJ!

DLA:	Dla Ciebie.
OD:	Ode mnie.
TEMAT:	Pisanie eleganckich notek.
DATA:	Dzisiaj.

Oto jeden ze sposobów tworzenia notki. Główną jej część wpisuje się tak jak każdy inny tekst w `TeX`-u: pusty wiersz oznacza nowy akapit, `\noindent` likwiduje wcięcie akapitu, itd.

Mam nadzieję, że jest to pouczająca notka.

```

1 \centerline{\bf XIII. KONWERSACJA}
2 \centerline{\sl Komedy warunkowe; prosty scenariusz.}
3 \headline{\hfil\it Przykład XIII\newpageno}
4
5 \newcount\spk
6 \def\beginscript {\bgroup \parindent=0pt \spk=1 \sl \rightskip,4in
7   \def\par {\ifnum\spk=1 \endgraf \it \spk=2 \leftskip,4in \rightskip0in
8     \else \endgraf \sl \spk=1 \leftskip0in \rightskip,4in \fi}}
9
10 \def\endscript {\egroup} \def\(#1){\rm #1}}
11
12 \bigskip\noindent Ta rozmowa pomiędzy Amerykaninem i~Kanadyjką została
13 podsłuchana podczas lotu liniami Iceland Air:\bigskip
14
15 \beginscript
16 \baselineskip=12pt \parskip=7,5pt
17 Nie wiem, po co ci tam w górze każą wam się uczyć dwóch języków --- my
18 świetnie sobie tutaj radzimy z~jednym.
19
20 Czy to nie okropne? To jest rzeczywiście niesprawiedliwe. Po co Anglicy
21 muszą znać francuski? Tak czy inaczej, oni wszyscy uczą się francuskiego,
22 ale spróbujmy znaleźć kogoś w~Quebecu, kto mówi po angielsku!
23
24 \{(Ze zrozumieniem.) To musi być ciężkie.\par
25 \{(Ciepło.) To interesujące. Czy był pan już kiedyś wcześniej w~Islandii?
26 \par Tak!\par Jak tam jest?\par
27 \{(W zadumie.) Ludzie tam są bardzo mili, bardzo bezpośredni, bardzo
28 uczciwi\} \dots \par
29 \{(Z~aprobatą.) Lubię to, lubię, kiedy ludzie są uczciwi. Ale tak jest w~całej
30 północnej Europie, prawda? Można zobaczyć tę różnicę we Francji. Na północy
31 wszyscy są rzeczywiście uczciwi, na południu okradają cię. Co do Włoch\} \dots \par
32 \{(Z~miną człowieka z~towarzystwa, światowca.) Tak! Polubi pani Islandczyków. Oni
33 byli prostymi ludźmi, ale zaczynają być bardziej cywilizowani --- nawet mają
34 Kentucky Fried Chicken\} \dots \par
35 \{(Ze spojrzeniem typu: „skoro dyskutujemy o~kulturze, to czuję grunt pod
36 nogami”.) Mówi się, że Islandia ma zdumiewającą sztukę współczesną.\par
37 \{(Trochę z~zakłopotaniem, nie bardzo orientując się w~temacie.)
38 Są również dobrzy w~sztuce projektowania mebli.\par
39 Muszę tam pojechać!\par Tak!\par
40 \endscript
41 \baselineskip=22 true pt \vfil\ejct

```

UWAGI

5: Komenda `\newcount` została omówiona na str. 68.
6–8: Komenda `\beginscript` rozpoczyna grupę (za pomocą `\bgroup`, komendy równoważnej `{`), w której zmieniono formatowanie akapitu. Koniec grupy wskazuje komenda `\egroup` wywołana przez `\endscript`; przywracane jest oryginalne formatowanie. Komendy `\bgroup` i `\egroup` są wykorzystane, ponieważ pozwalają stworzyć grupę, która obejmuje definicje, nawet jeśli bezpośrednie użycie nawiasów `{ i }` nie zezwala na to. Nowy format akapitu powstał przez zdefiniowanie zwykłej komendy końca akapitu `\par`: krój przełączany jest pomiędzy kursywą i pochyłym w kolejnych akapitach i są one wcinane na przemian, raz z lewej, raz z prawej, dzięki komendom `\leftskip` i `\rightskip`. Zmienna zawarta w rejestrze `\spk` oznacza akapity jako „1” bądź „2”. Kombinacja komend `\ifnum... \else... \fi` wybiera jeden z forma-

tów, gdy `\spk` oznacza „1”, drugi zaś — w przeciwnym wypadku. `\fi` musi być wykorzystana do zakończenia wszystkich komend warunkowych, z wyjątkiem komend wewnątrz `\loop`. `\endgraf` jest standardową komendą, dającą efekt taki jak zwykła komenda `\par`. Jest ona przydatna wtedy, kiedy `\par` jest tymczasowo zdefiniowana.

10: `\(#1)` jest komendą bez liter w nazwie. W wierszu 24 znajduje się przykład jej zastosowania.

19, 23: Pusty wiersz jest znakiem końca akapitu, co powoduje zmianę formatu akapitu *bez potrzeby używania jakichkolwiek komend dodatkowych*.

24, 28, itd.: Akapity mogą być również zakończone zwykłą komendą `\par`.

26: Nowy akapit nie musi rozpoczynać się w osobnym wierszu, ale taki sposób aranżowania pliku wejściowego jest mało czytelny.

XIII. KONWERSACJA

Komendy warunkowe; prosty scenariusz.

Ta rozmowa pomiędzy Amerykaninem i Kanadyjką została podsłuchana podczas lotu liniami Iceland Air:

Nie wiem, po co ci tam w górze każą wam się uczyć dwóch języków — my świetnie sobie tutaj radzimy z jednym.

Czy to nie okropne? To jest rzeczywiście niesprawiedliwe. Po co Anglicy muszą znać francuski? Tak czy inaczej, oni wszyscy uczą się francuskiego, ale spróbujmy znaleźć kogoś w Quebecu, kto mówi po angielsku!

Ze zrozumieniem. To musi być ciężkie.

Ciepło. To interesujące. Czy był pan już kiedyś wcześniej w Islandii?

Tak!

Jak tam jest?

W zadumie. Ludzie tam są bardzo mili, bardzo bezpośredni, bardzo uczciwi...

Z aprobatą. Lubię to, lubię, kiedy ludzie są uczciwi. Ale tak jest w całej północnej Europie, prawda? Można zobaczyć tę różnicę we Francji. Na północy wszyscy są rzeczywiście uczciwi, na południu okradają cię. Co do Włoch...

Z miną człowieka z towarzystwa, światowca. Tak! Polubi pani Islandczyków. Oni byli prostymi ludźmi, ale zaczynają być bardziej cywilizowani — nawet mają Kentucky Fried Chicken...

Ze spojrzeniem typu: „skoro dyskutujemy o kulturze, to czuję grunt pod nogami”. Mówi się, że Islandia ma zdumiewającą sztukę współczesną.

Trochę z zakłopotaniem, nie bardzo orientując się w temacie. Są również dobrzy w sztuce projektowania mebli.

Muszę tam pojechać!

Tak!

```

1 \centerline{\bf XIV. OSTATNIE TWIERDZENIE FERMATA}
2 \centerline{\sl Programowanie w\TeX-u.}
3 \headline{\hfil\it Przykład XIV\newpageno}\medskip%\newpageno --- KOMENDA WŁASNA
4
5 \newcount\var \newcount\pw \newcount\tmp \newcount\cnt
6 \def\pow#1#2#3{\var=#1 \pw=#2 \tmp=#3 \cnt=1
7   \loop \multiply\var by\tmp \advance\cnt by1 \ifnum\cnt<\pw \repeat
8   \global#3=\var}
9 \newcount\xf \newcount\xnf \newcount\yf \newcount\ynf \newcount\zf \newcount\znf
10 \def\n {\number}
11
12 \def\fermat#1#2#3{ $\global\xf=#1 \global\yf=#2 \global\pw=#3
13   \pow{\xf}{\pw}{\xnf} \pow{\yf}{\pw}{\ynf}
14   \global\tmp=\ynf \global\advance\tmp by\xnf
15   {\n\xf}^{\n\pw}+{\n\yf}^{\n\pw}={\n\tmp}$.\hfil\break
16   \ifnum\xf>\yf \zf=\xf \else \zf=\yf \fi
17   \loop {\pow{\zf}{\pw}{\znf}} \ifnum\znf<\tmp \advance\zf by1 \repeat
18   \ifnum\znf=\tmp Suma będzie dokładnie ${\n\zf}^{\n\pw}$.
19   \ifnum\pw=2 {\it Ziewnięcie} Powiedz mi coś, czego nie wiem, dobrze?
20   \else Niemożliwe! Ale może lepiej sprawdź moją pracę.\fi
21   \else \advance\zf by-1
22   Liczba ta leży pomiędzy ${\n\zf}^{\n\pw}$\pow{\zf}{\pw}{\znf}($={\n\znf}$)
23   \advance\zf by1
24   i ${\n\zf}^{\n\pw}$\pow{\zf}{\pw}{\znf}($={\n\znf}$).\fi}%
25
26 \noindent Czy dla $n>2$ istnieją dodatnie liczby całkowite $x$, $y$, $z$ i $n$,
27 spełniające równanie $x^n + y^n = z^n$? Pierre Fermat, ponad trzysta lat temu,
28 twierdził, iż może udowodnić, że nie. Nigdy nie znaleziono jego dowodu ani nikt
29 nie odkrył do tej pory innego. By stwierdzić prawdziwość twierdzenia
30 Fermata, popatrzmy na kilka przykładów:
31 \medskip
32 \item{\it 1.}\fermat{73}{71}{3} \smallskip
33 \item{\it 2.}\fermat{2}{3}{13} \smallskip
34 \item{\it 3.}\fermat{23}{37}{5} \medskip
35 \noindent („Fascynujące”, wyszeptales, „bez wątpienia wyczerpujące,
36 ale jaką głęboką \TeX-ową prawdę to ilustruje?” Jesteśmy teraz przy
37 przykładzie XIV i~wiesz już, jaki plik wejściowy tworzy coś takiego. Przeczytaj
38 uwagi na sąsiedniej stronie, dokonaj analizy pliku wejściowego i~podziwiał
39 możliwości \TeX-a.) \vfil\ejct

```

UWAGI

KOMENTARZ

Ten przykład i wyjaśnienia poniżej nie będą łatwe do zrozumienia, jeśli nie zna się nieco programowania lub przynajmniej matematyki. Przykład ten pokazuje, jak można wykorzystać \TeX -a do kształtowania zawartości dokumentu, a nie tylko jego formy. Wiersze 5–24 definiują nowe komendy, z których główną jest `\fermat`. Wiersze 32–34 pokazują, jak można wykorzystać tę komendę: należy podać trzy liczby, reprezentujące x , y i n . Program wewnętrznie liczy wartość wyrażenia $x^n + y^n$, znajduje najbliższą n -tą potęgę tej sumy i wyświetla starannie złożoną odpowiedź.

5, 9: Rejestry te będą potrzebne do obliczeń. Komenda `\newcount` była wyjaśniona na str. 68.

6–8: `\pow` pobiera parametr **#1**, podnosi go do potęgi **#2** i odpowiedź przyrównuje do **#3**. `\global` jest wykorzystywana do poprzedzania przypisań i definicji, które mają działać także poza grupą.

10: `\number` składa znaki reprezentujące liczbę. Będzie wykorzystana w kilku miejscach do wydrukowania zawartości rejestru, stąd wprowadzony jest skrót `\n`.

12–15: Program liczy n -tą potęgę z `\pow`, a następnie znajduje sumę tych potęg (i umieszcza ją w `\tmp`). Znaki `$` nie są konieczne do obliczeń, ale są potrzebne do składu zawartości wiersza 15. Wykonują dodatkowo usuwanie odstępów, które czasami są wprowadzane do pliku wyjściowego przez skomplikowane komendy.

16: Większe z wartości x i y są wartościami początkowymi dla z .

17: n -ta potęga kolejnych wartości z jest obliczana do momentu, gdy będzie równa lub większa od liczby w `\tmp`.

22–24: n -te potęgi poniżej i powyżej `\tmp` są natychmiast liczone i przedstawiane.

XIV. OSTATNIE TWIERDZENIE FERMATA

Programowanie w T_EX-u.

Czy dla $n > 2$ istnieją dodatnie liczby całkowite x , y , z i n , spełniające równanie $x^n + y^n = z^n$? Pierre Fermat, ponad trzysta lat temu, twierdził, iż może udowodnić, że nie. Nigdy nie znaleziono jego dowodu ani nikt nie odkrył do tej pory innego. By stwierdzić prawdziwość twierdzenia Fermata, popatrzmy na kilka przykładów:

1. $73^3 + 71^3 = 746928$.

Liczba ta leży pomiędzy 90^3 (= 729000) i 91^3 (= 753571).

2. $2^{13} + 3^{13} = 1602515$.

Liczba ta leży pomiędzy 3^{13} (= 1594323) i 4^{13} (= 67108864).

3. $23^5 + 37^5 = 75780300$.

Liczba ta leży pomiędzy 37^5 (= 69343957) i 38^5 (= 79235168).

(„Fascynujące”, wyszeptaleś, „bez wątpienia wyczerpujące, ale jaką głęboką T_EX-ową prawdę to ilustruje?” Jesteśmy teraz przy przykładzie XIV i wiesz już, jaki plik wejściowy tworzy coś takiego. Przeczytaj uwagi na sąsiedniej stronie, dokonaj analizy pliku wejściowego i podziwiaj możliwości T_EX-a.)


```

1 \centerline{\bf XV.\quad\hbox{\raise6pt\hbox{0}
2 \kern8pt\raise6pt\hbox{B}\raise3pt\hbox{0}K\lower3pt\hbox{S}%
3 \lower6pt\hbox{IE}}}}
4 \centerline{\sl Umieszczanie materiału w pionowych i poziomych pudełkach.}
5 \headline{\hfil\it Przykład XV\newpageno} \bigskip \baselineskip=14pt
6 \hbox to\hsize{\leaders\hbox to 20pt{\ast$\hfil}\hfill}
7 \hbox to\hsize{\leaders\hbox to 20pt{\hfil$\ast$\hfil}\hfill}
8 \hbox to\hsize{\leaders\hbox to 20pt{\hfil$\ast$\hfil}
9 \vfil
10 \vbox{\hsize 5.2 true in Podczas pierwszorzędnego meczu bokserskiego
11 (np. Ali-Frazier~I) jesteŃmy głęboko poruszeni wspólnotą ciała z nim samym,
12 udającym inne bezkompromisowe ciało.\hfil\break
13 \hbox to\hsize{\hfill Joyce Carol Oates, {\it On Boxing}, Doubleday (1987).}}
14 \vfil
15 \moveright ,5 true in\vbox{\hbox{\dots Ali był zmęczony.}
16 \hbox{Uderzył Foremana mocniej, niż kiedykolwiek uderzył kogoś.}
17 \hbox{Często go uderzał.} \hbox{Głowa Foremana musi być podobna teraz
18 do kawałka wulkanizowanej gumy.}
19 \hbox{\hfill Norman Mailer, {\it The Fight}, Little Brown (1975).}}\vfil
20 \hbox{\vbox{\hsize 3,3 true in \baselineskip=14 true pt
21 Urazy głowy stanowiły 68 procent kontuzji. Do tej kategorii należały: krwotok
22 móz\~gu, pęknięcie czaszki oraz złamania kości twarzy. W następnej kolejności
23 są uszkodzenia górnych kończyn i tułowia.
24
25 R.W. Enzenauer i in., \sl Boxing-related injuries in the U.S. Army, \it
26 The Journal of the American Medical Assoc., \rm March 10, 1989.}
27 \kern ,2 true in
28 \vbox{\hsize 2,6 true in Mogę przeklinać plamy krwi na mojej koszuli\dots\
29 wciąż jednak wracam i~wracam\dots\ Czy to robi ze mnie humanistę, czy zbrojnicę,
30 nie jes\~tem pewien.\hfil\break John Schullian,\ \ {\it Writers' Fighters\break
31 and Other Sweet Scientists}, \hfil\break Andrews and McMeel (1983).}} \vfil
32 \hbox to\hsize{\leaders\hbox to 20pt{\hfil$\smile$\hfil$\frown$\hfill}
33 \hbox to\hsize{\hfil
34 \vbox to,5in{\hsize ,2in\leaders\vbox to 8pt{\vfil\noindent$\diamond$\par}
35 \vfill} \quad
36 \vbox to,5in{\hsize ,2in\leaders\vbox to
37 8pt{\vfil\noindent$\diamond$\par\vfil} \vfill} \quad \vbox to,5in{\hsize
38 ,2in\leaders\vbox to 8pt{\noindent$\diamond$\par\vfil} \vfill}\hfil}
39 \baselineskip=22 true pt \eject

```

UWAGI

O PUDEŁKACH

Pudełka (*boxes*) pozwalają na pełną kontrolę nad składanym materiałem. `\hsize` i `\vsize` to rozmiary w poziomie (szerokość) i w pionie (wysokość). Zewnętrznie wprowadzone są wymiarami strony, wewnętrznie — rozmiarami pudełka. `\hbox` najczęściej ma szerokość zawartości pudełka; podobnie jak `\vbox` — wysokość. Te rozmiary mogą być zmieniane np. za pomocą wpisu umieszczanego zaraz za nazwą pudełka „to nowy rozmiar” (wiersze 6–8 lub 32–38). Poziomy rozmiar `\vbox`-a może zostać wyspecyfikowany wewnątrz pudełka (jak w wierszu 10) lub może być podany w sposób domniemany: `\vbox` zawierający `\hbox` przyjmie szerokość `\hbox`-a, a w przypadku gdy wymiar ten nie jest określony, `\vbox` będzie miał szerokość strony. Pudełka mogą być przemieszczane w górę lub w dół za pomocą komend `\raise` lub `\lower` albo w prawo lub w lewo przy użyciu instrukcji `\moveright` lub `\moveleft`. Pudełko może być przemieszczane pionowo tylko wtedy, gdy TeX przetwarza materiał *poziomo* (tzn. składa wiersze), po-

ziomo zaś — przy przetwarzaniu *pionowym*.

1–3: `\raise` i `\lower` przemieszczają pudełka. `\kern` w tym kontekście składa odstępy poziome.

6–8: `\hsize` w tym miejscu jest szerokością strony (kolumny tekstu). `\leaders` to tzw. wypełniacz. Wypełnia znakiem * 20-punktowe pudełko. Komendy `\hfil` i `\hfill` kontrolują sposób wypełnienia (przesunięcie).

13: Ponieważ wiersz ten znajduje się wciąż wewnątrz pudełka `\vbox` rozpoczętego w linii 10, `\hsize` określony w tym miejscu nie jest rozmiarem strony, jak to wykorzystano domyślnie w wierszach 6–8.

15: `\moveright` przesuwają pudełko w prawo.

15–19: `\hbox`-y wewnątrz `\vbox`-a składane są jedno na drugim. `\hfill` w wierszu 19 nie wpływa na skład, ponieważ `\hbox` ma rozmiar zawartości.

20–31: `\vbox`-y wewnątrz `\hbox`-a składane są jedno obok drugiego. `\kern` składa poziomy odstępy między pudełkami.

34–38: `\leaders` składa wzór w pionie.

XV. O BOKSIE

Umieszczanie materiału w pionowych i poziomych pudełkach.

```
* * * * *
* * * * *
* * * * *
```

Podczas pierwszorzędnego meczu bokserskiego (np. Ali-Frazier I) jesteśmy głęboko poruszeni wspólnotą ciała z nim samym, udającym inne bezkompromisowe ciało.

Joyce Carol Oates, *On Boxing*, Doubleday (1987).

... Ali był zmęczony.

Uderzył Foremana mocniej, niż kiedykolwiek uderzył kogoś.

Często go uderzał.

Głowa Foremana musi być podobna teraz do kawałka wulkanizowanej gumy.

Norman Mailer, *The Fight*, Little Brown (1975).

Urazy głowy stanowiły 68 procent kontuzji. Do tej kategorii należały: krwotok mózgu, pęknięcie czaszki oraz złamania kości twarzy. W następnej kolejności są uszkodzenia górnych kończyn i tułowia.

R.W. Enzenauer i in., *Boxing-related injuries in the U.S. Army*, *The Journal of the American Medical Assoc.*, March 10, 1989.

Mogę przeklinać plamy krwi na mojej koszuli... wciąż jednak wracam i wracam... Czy to robi ze mnie humanistę, czy zbrojeńca, nie jestem pewien.

John Schulian, *Writers' Fighters and Other Sweet Scientists*, Andrews and McMeel (1983).

```
~ ~ ~ ~ ~
  ◇   ◇   ◇
  ◇   ◇   ◇
  ◇   ◇   ◇
  ◇   ◇   ◇
```

```

1 \def\frame #1#2#3#4{\vbox{\hrule height #1pt% GÓRNA KRESKA RAMKI
2 \hbox{\vrule width #1pt\kern #2pt% KRESKA I ODSTĘP LEWOSTRONNY
3 \vbox{\kern #2pt% ODSTĘP OD GÓRY DO ZAWARTOŚCI
4 \vbox{\hspace #3\noindent #4}% ZAWARTOŚĆ PUDEŁKA
5 \kern #2pt}% ODSTĘP DOLNY
6 \kern#2pt\vrule width #1pt}% ODSTĘP I KRESKA PRAWOSTRONNA
7 \hrule heightOpt depth #1pt}}% DOLNA KRESKA RAMKI
8
9 \def\fitframe #1#2#3{\vbox{\hrule height#1pt% GÓRNA KRESKA RAMKI
10 \hbox{\vrule width#1pt\kern #2pt% KRESKA I ODSTĘP LEWOSTRONNY
11 \vbox{\kern #2pt\hbox{#3}\kern #2pt}% ODSTĘP GÓRNY, ZAWARTOŚĆ, ODSTĘP DOLNY
12 \kern #2pt\vrule width#1pt}% KRESKA I ODSTĘP PRAWOSTRONNY
13 \hrule heightOpt depth#1pt}}% DOLNA KRESKA RAMKI
14
15 \def\shframe #1#2#3#4{\vbox{\hrule height Opt% GÓRA RAMKI BEZ CIENIA
16 \hbox{\vrule width #1pt\kern Opt% LEWY CIENŃ
17 \vbox{\kern-#1pt\frame{,3}{#2}{#3}{#4}% POCZĄTEK CIENIA #1pt OD GÓRY
18 \kern-,3pt}% KOREKTA POŁOŻENIA DOLNEGO CIENIA
19 \kern-#1pt\vrule width Opt}% KONIEC DOLNEGO CIENIA O 1#pt OD KOŃCA
20 \hrule height #1pt}}% DOLNY CIENŃ
21
22 \def\s #1{\frame{,3}{2}{8pt}{\centerline{#1\vphantom{()}}}\ }
23
24 \centerline{\bf XVI. RUCH DOOKOŁA}
25 \centerline{\sl Obramowanie tekstu; cieniowanie pudełek; inne ciekawe efekty.}
26 \headline{\hfil\it Przykład XVI\newpageno}% \newpageno --- KOMENDA WŁASNA
27 \vfil
28 \centerline{\frame{,1}{10}{3in}{\noindent Obramowanie tekstu w pudełkach jest
29 często jedynym skutecznym sposobem zwrócenia uwagi czytelników. Istotny jest
30 jednak fakt, czy mamy czytelnikom coś interesującego do powiedzenia.}}
31 \vfil
32 \centerline{\fitframe{2}{1}{\frame{,8}{6}{2,2in}{\noindent Cokolwiek robisz,
33 nie wkładaj materiału w obramowane pudełka tylko w celu jego odgrodzenia.}}
34 \vfil
35 \centerline{\fitframe{,3}{,5}{Ciężko jest unikać takich przyjemnych dopasowań!}}
36 \vfil
37 \centerline{\shframe{3}{10}{130pt}{\centerline{I na koniec, to powinno}}}
38 \vskip 30 pt
39 \centerline{\shframe{12}{10}{120pt}{\centerline{ZASKOCZYĆ CIĘ.}}}\quad
40 \vfil\eject

```

UWAGI

1–7: `\frame` składa ramkę (pudełko z kreski o wyspecyfikowanej szerokości) wokół tekstu. **#1** jest grubością kreski w punktach, **#2** — odległością pomiędzy kreskami a obramowanym tekstem (w punktach), **#3** — szerokością pudełka (w określonych jednostkach), a **#4** — faktycznym tekstem. Jeśli tekst, który chcemy obramować, składa się z kilku akapitów, należy podać „`\long\def`” zamiast `\def`. (Standardowo \TeX nie pozwala na umieszczenie więcej niż jednego akapitu w miejscu parametru komendy.) `\kern` składa odstęp poziomy albo pionowy, w zależności od kontekstu. Znak % pozwala na umieszczenie komentarza objaśniającego.

9–13: `\fitframe` reguluje poziomy rozmiar, by dopasować go do zawartości. **#1** i **#2** to znowu, odpowied-

nio, grubość kreski i odstęp pomiędzy kreską a tekstem, **#3** jest tekstem. Ten tekst powinien nie być dłuższy niż poziomy rozmiar strony, jeśli nie ma się rozlewać na marginesy.

15–20: `\shframe` składa ramkę „z cieniem”. **#1** jest grubością cienia w punktach, a **#2**, **#3** i **#4** są identyczne jak w przypadku `\frame`. Można dowolnie zmieniać wartości poszczególnych parametrów, przemieszczać cień, zmieniać `\kern`.

22: `\s` składa pudełko na wysokość jednej litery; jest to `\frame` o rozmiarze poziomym 8 punktów z wycentrowaną zawartością. Efekt działania tej komendy można zobaczyć na str. 79 i 83.

28–40: Kilka sposobów wykorzystania komend dotyczących ramek.

XVI. RUCH DOKOŁA

Obramowanie tekstu; cieniowanie pudełek; inne ciekawe efekty.

Obramowanie tekstu w pudełkach jest często jedynym skutecznym sposobem zwrócenia uwagi czytelników. Istotny jest jednak fakt, czy mamy czytelnikom coś interesującego do powiedzenia.

Cokolwiek robisz, nie wkładaj materiału w obramowane pudełka tylko w celu jego odgrózenia.

Cieężko jest unikać takich przyjemnych dopasowań!

I na koniec, to powinno

ZASKOCZYĆ CIĘ.

```

1 \newcount\tw      % g6rna podstawa w pt
2 \newcount\bw      % dolna podstawa w pt
3 \newcount\h       % wysoko6c w pt
4 \newcount\bs      % przesuni6cie lewego g6rnego wierzcho6ka w pt
5 \newcount\th      % grubo6c kreski w 1/64 pt
6 \newcount\gp      % odleglo6c mi6dzy kreskami w 1/64 pt
7 \newcount\rs      % ruchome przesuni6cie
8 \newcount\rw      % ruchoma szeroko6c
9 \newcount\rh      % ruchoma wysoko6c
10 \newcount\tmp     % zmienna tymczasowa (pomocnicza)
11
12 \def\trap #1#2#3#4#5#6{\vbox{\offinterlineskip
13   \tw=#1 \bw=#2 \h=#3 \bs=#4 \th=#5 \gp=#6 \rh=0
14   \multiply\tw by 65536 \multiply\bw by 65536 \multiply\bs by 65536
15   \multiply\th by1024 \multiply\gp by1024
16   \loop
17   \tmp=\bs \multiply\tmp by\rh \divide\tmp by\h
18   \rs=\tmp %obliczanie ruchomego przesuni6cia
19   \tmp=\bw \advance\tmp by-\tw \multiply\tmp by\rh
20   \divide\tmp by\h \advance\tmp by\tw \rw=\tmp%obliczanie ruchomej szeroko6ci
21   \hbox{\kern\rs sp\vrule height0sp depth\th sp width\rw sp}%RYSOWANIE KRESEK
22   \vskip\gp sp %ODST6P MI6DZY KRESKAMI
23   \ifnum\rh<\h \tmp=\rh \multiply\tmp by 65536
24   \advance\tmp by\th \advance\tmp by\gp \divide\tmp by65536 \rh=\tmp
25   \repeat}}
26
27 \def\rect #1#2#3#4{\trap{#1}{#1}{#2}{0}{#3}{#4}}
28
29 \line{\hfil}
30 \vfil
31 \centerline{\s{D}\s{L}\s{A}\quad
32 \s{U}\s{K}\s{O}\s{R}\s{O}\s{N}\s{O}\s{W}\s{A}\s{N}\s{I}\s{A}\s{.}\s{.}\s{.}}
33 \vfil
34 \vbox{\offinterlineskip
35 \centerline{\trap{0}{60}{180}{30}{10}{60}%
36 \trap{2}{65}{100}{-25}{20}{60}\hskip20pt%
37 \trap{2}{65}{100}{-38}{20}{60}\hskip65pt%
38 \trap{0}{60}{180}{-90}{10}{60}}
39 \centerline{\rect{200}{15}{5}{100}}
40 \vfil\vfil\eject

```

UWAGI

1–10: Zmienne te s1 zmiennymi pomocniczymi.

12–25: `\trap` sk1ad1 trapezoid zbudowany z kresk poziomych. #1 i #2 to jego podstawy, #3 — wysoko6c, a #4 — przesuni6cie lewego g6rnego wierzcho6ka wzgl6dem lewego wierzcho6ka podstawy. Wszystkie parametry podane s1 w punktach. #5 jest grubo6ci1 kreski a #6 — pionow1 odleglo6ci1 mi6dzy kolejnymi kreskami; oba wymiary w 1/64. cz6ci punktu. Wyb6r zbyt ma6lych warto6ci tych parametr6w spowoduje wy6wietlenie informacji o b6ledzie. W trakcie rysowania kreski, w odpowiednich rejestrach umieszczana jest bie611ca wysoko6c trapezoidu `\rh` oraz warto6ci ruchomej szeroko6ci `\rw` i przesuni6cia `\rs`.

12, 34: `\offinterlineskip` likwiduje odst6py mi6dzy wierszami, tak wi6c ka6d1 kolejna rysowana kreska znajdzie si6 bezpo6rednio na szczycie poprzedniej, bez dodatkowego odst6pu.

13: Ka6dzy z nowo oznaczonych rejestr6w odnosi si6 do odpowiedniej zmiennej wej6ciowej.

14–15: T6X zaokr1gla zmienne do najbli6szych liczb ca6kowitych, nie nale6y wi6c stosowa6 ma6lych jednostek. Liczby tu wybrane powoduj1, 6e wszystkie

zmienne wykorzystuj1 jednostki *skalowanych punkt6w* (65536 skalowanych punkt6w to 1 punkt).

17–18: „Ruchome przesuni6cie” (*running shift*) liczy si6 przez pomno6zenie przesuni6cia g6rnego wierzcho6ka `\bs` przez warto6c ilorazu bie611cej wysoko6ci `\rh` i wysoko6ci `\h`.

19–20: Szeroko6c kolejnej rysowanej kreski jest obliczona podobnie.

21: Rysowana kreska jest teraz przemieszczona (za pomoc1 `\kern`) o aktualn1 warto6c przesuni6cia.

22: W tym miejscu jest zostawiony odst6p mi6dzy kreskami.

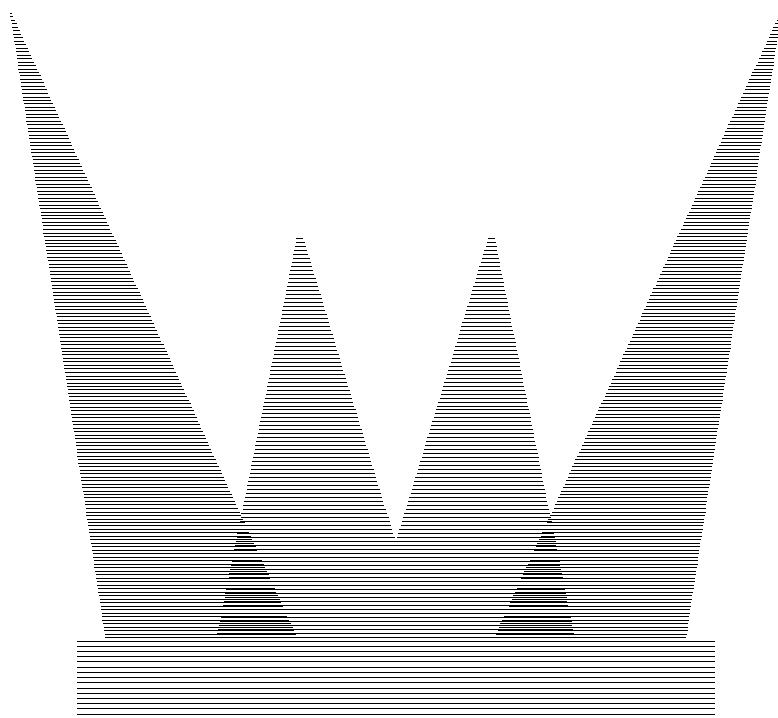
23–25: Je6li bie611ca wysoko6c wci16z nie jest wysoko6ci1 ko6ncow1, zostaje zwi6kszona o grubo6c kreski plus odst6p mi6dzy kreskami, i ca6y proces jest powtarzany.

27: `\trap` s6u6y do narysowania czwor6k1ta o szeroko6ci #1, wysoko6ci #2, z kresk o grubo6ci #3, z odleglo6ci1 mi6dzy kreskami #4.

31–32: Efekt dzia6ania komendy `\s`, zdefiniowanej na str. 76.

35–39: Efekt dzia6ania komend `\trap` i `\rect`.

D L A U K O R O N O W A N I A . . .



```

1 \centerline{\bf XVII. SCENY Z CZASOPISMA}
2 \centerline{\sl Proste aspekty składu czasopism.}
3 \headline{\hfil\it Przykład XVII\newpageno} \bigskip \hrule \bigskip
4
5 \footline{\sl \ifnum\pageno=1 \hfil \else
6 \ifodd\pageno Data \hfil \the\pageno \else \the\pageno \hfil \sl BRUTAL\fi\fi}
7
8 \newcount\colnumber \newbox\col \newdimen\tmpdim \newdimen\size
9 \newdimen\coljump \coljump=,2 true in %odl. między kolumnami.
10
11 \def\niceskip {\vskip\baselineskip}
12 \def\nstrut {\vrule height \topskip depth0pt width0pt}%
13
14 \def\beginart #1/#2{\vbox\bgroup#1 \colnumber=#2 \parskip=0pt
15 \advance\colnumber by-1 \tmpdim=\coljump \multiply\tmpdim by\colnumber
16 \size=\hsize \advance\size by-\tmpdim
17 \advance\colnumber by1 \divide\size by\colnumber
18 \vbadness=10000 \hbadness=2000 \tolerance=2000
19 \setbox\col=\vbox\bgroup\hsize\size \noindent\nstrut}
20
21 \def\endart {\global\size=\baselineskip \vfil \egroup
22 \multiply\size by\colnumber \advance\size by-\baselineskip
23 \tmpdim=\ht\col \advance\tmpdim by\size \divide\tmpdim by\colnumber
24 \hbox{\splittopskip=\topskip \doittoit}\egroup}
25
26 \def\doittoit{\ifnum\colnumber>0 \vsplit\col to \tmpdim
27 \global\advance\colnumber by-1
28 \ifnum\colnumber>0 \hskip\coljump \fi \doittoit \fi}
29
30 \font\head=plinch scaled 700 \font\shead=pldunh10 scaled\magstep2
31 \font\blare=plss10 scaled 2073 \font\title=plssdc10 scaled 2488
32
33 \centerline{\vbox{\line{\head B\hfil R\hfil U\hfil T\hfil A\hfil L}
34 \bigskip \hrule \medskip
35 \line{\shead \ Magazyn \hfil Wrażliwego \hfil Mężczyzny\ }
36 \smallskip \hrule}} \vfil
37 \centerline{\title Numer specjalny} \vfil
38 \centerline{\title KOMPUTERY} \vskip ,3in
39 \centerline{\title \&}\vskip ,3in
40 \centerline{\title SZTUKA KULINARNA} \vfil
41 \centerline{\blare wspomnienia BRUTALA,} \bigskip
42 \centerline{\blare czyli przepisy z ostatnich 15 lat} \vfil\hrule\eject

```

UWAGI

5–6: Bieżący numer strony umieszczony jest w rejestrze oznaczonym `\pageno`. Wiersze te ustawiają `\footline`: jeśli numer strony wynosi 1, stopka jest pusta; w przeciwnym wypadku numery stron parzystych i nieparzystych składane są po przeciwnych stronach.

8–9: Znajdują się tu rejestry `\box` i `\dimen`, które będą wykorzystane do komend `\beginart` i `\endart`. `\coljump` stanowi odstęp pomiędzy kolumnami; jest to wartość podana wprost.

11: `\niceskip` składa pionowy odstęp, równy odległości między wierszami.

12: `\topskip` jest odstępem pozostawionym przez program między szczytem pudełka a linią podstawową pierwszego wiersza.

14–19: `\beginart` określa początek materiału do artykułu, który ma być przełamany na `#2` kolumny. Dla parametru `#1` można podać kilka wielkości: nowe `\hsize`, `\coljump`, itd. Przykład zastosowania tej

komendy przedstawiono na str. 84. Komendy w wierszach 15–17 pozwalają na obliczenie szerokości kolumny, z uwzględnieniem odległości między kolumnami. Komenda w wierszu 18 zmienia parametry służące do kontroli nad odstępami; komendy te omówione są w dodatku.

21–24: `\endart` wskazuje koniec materiału do artykułu i oblicza wysokość kolumn (zawartą w `\tmpdim`). Kolumny umieszczane są w pudełku poziomym `\hbox`, a `\splittopskip` określa odstęp nad każdą z kolumn. Podział na kolumny został dokonany za pomocą komendy `\doittoit`.

26–28: `\vsplit` odłącza od pudełka z gromadzonym materiałem część na szczycie o wysokości `\tmpdim`. Do wszystkich kolumn, poza ostatnią, dołączany jest odstęp poziomy `\coljump`. Operacja ta jest powtarzana (poprzez rekurencję) aż do wyczerpania liczby kolumn; definicja `\doittoit` wywołuje się sama.

30–31: Definicje krojów pisma.

XVII. SCENY Z CZASOPISMA

Proste aspekty składu czasopism.

BRUTAL

Magazyn

Wrażliwego

Mężczyzny

Numer specjalny

KOMPUTERY

&

SZTUKA KULINARNA

wspomnienia BRUTALA,
czyli przepisy z ostatnich 15 lat


```

1 \def\ff #1{\fitframe{,2}{,7}{#1}}
2
3 \hrule
4 \vskip 1,2 in
5 \centerline{\vbox{\hsize260pt
6 \centerline{\fitframe{5}{,7}{\ff{\ff{\ff{\ff{\ff{\frame{,2}{10}{160pt}}%
7 {\vskip15pt \blare
8 \centerline{Komputerowe Rady}
9 \medskip
10 \centerline{z}
11 \bigskip
12 \centerline{\title BRUTALA}
13 \vskip15pt}}}}}}}}
14 \medskip
15 \vskip1pt
16 \hbox{\frame{,5}{7}{245pt}{\line{\quad
17 \vbox{\hbox to1,5in{\leaders\hbox to 5pt{\$ \rangle$ \hfill} \hfill}} \hfil
18 \vrule height2pt depth 0pt width40pt \hskip,3in}}}
19 \bigskip
20 \vskip1pt
21 \hbox{\frame{,5}{7}{245pt}{\sans
22 \centerline{\ \s{} \s{} \s{} \s{J} \s{A} \s{K} \s{} \s{} \s{} \s{}}
23 \vskip2pt
24 \centerline{\ \s{} \s{K} \s{0} \s{N} \s{S} \s{E} \s{R} \s{W} \s{0} \s{W} \s{A} \s{Ć} \s{}}
25 \vskip2pt
26 \centerline{\ \s{} \s{T} \s{W} \s{A} \s{R} \s{D} \s{Y} \s{} \s{D} \s{Y} \s{S} \s{K}}
27 \vskip4pt
28 \centerline{\fitframe{,3}{2}{\ \rm Nasz Stały Felietonista\ }}}}
29 \vfil
30 \hrule\eject

```

UWAGI

KOMENTARZ

Komendy dotyczące ramek, wykorzystane na tej stronie, były zdefiniowane w przykładzie XVI.

1: Skrót dla `\fitframe` składający kreskę o grubości 0,2 punktu i odstępem pomiędzy kreską a tekstem 0,7 punktu.

5: `\vbox` rozpoczynający się w tym miejscu będzie zamknięty w linii 28; cały rysunek na tej stronie będzie

traktowany jako pojedyncza jednostka.

6–13: Komendy w tych wierszach „rysują monitor”.
6: Ramki są dopasowywane kolejno, jedna wewnątrz drugiej.

7, 12: `\blare` i `\title` to nazwy krojów pisma, zdefiniowanych na poprzedniej stronie.

21–28: Komendy znajdujące się w tych wierszach składają „klawiaturę”.


```

1 \hrule\vskip1,2in
2 \font\tinyrm=plr7
3 \leftline{\blare Szafranowe lody}
4 \smallskip
5 \hrule height1pt depth0pt width\hsize
6 \smallskip
7 \rightline{\sans kącik dla kucharzy}
8 \bigskip
9 \beginart \hsize=5,4 true in \coljump=,15 true in / {3}
10 \baselineskip=14 true pt \tinyrm
11 Szafran to przyprawa z~suszonych kwiatów krokusa, należącego do rodziny irysów.
12 Ma wspaniałe zapach i~smak, nie posiadające substytutów; to jedna z~najdroższych
13 przypraw w~świecie --- sprzedawana w~cenie cztery razy niższej od ceny złota.
14 Mała porcja tej przyprawy wystarcza na długo, dlatego czasem mo~żna
15 się na nią skusić.
16
17 Dzięki temu przepisowi można zrobić niezapomniane lody. Nawet najmniej
18 zainteresowani jedzeniem przerwą rozmowy, ich oczy rozszerzą się
19 z~zaciekawieniem, kiedy pierwsza łyżeczka lodów dotknie ich języka.\niceskip
20 \centerline{\bf Składniki}
21 \item{1/3} filiżanki śmietany.
22 \item{1} duża szczypta włosków szafranu (ok. 0,5 g).
23 \item{12} torebek kardamonu.
24 \item{2,3 1} lodów waniliowych.\niceskip
25 \centerline{\bf Wykonanie} \noindent
26 Podgrzać śmietanę i~dodać szafran. Pozostawić ją na bardzo wolnym ogniu, na
27 godzinę. Jeśli wydaje się, że zaczyna parować, dodać więcej śmietany. Śmietana
28 przybierze intensywnie złoty kolor. Następnie obrać i~rozgnieść nasiona
29 kardamonu, dodać do śmietany, wymieszać wszyst~ko i~zdjąć z~ognia.\par
30 Kilka minut przed podaniem umie~ścić lo~dy w du~zym pu~cha~rze i~polać
31 miksturą ze śmietany. Za po~mo~cą rę~czne~go miksera ubić śmietanę
32 z lodami. Wymaga to siły fizycznej i~dużej uwagi, ale zapłata za wysiłek już
33 jest bardzo blisko. Mieszać należy od brzegu naczynia (łatwiej to robić, jeśli
34 lody nie są bardzo zmrożone).\par Podawać od razu.
35 \endart
36 \bigskip
37 \hrule height 1pt depth0pt width\hsize
38 \vfil\hrule\eject
39 \footline{\hfil}

```

UWAGI

Przedstawiona tu kombinacja komend `\beginart...` `\endart` jest niestety odpowiednia tylko w przypadku krótkich artykułów, które można złożyć w obrębie jednej strony.

9: Komendy przed znakiem `/` są opcjonalne, ale pokazują, jak wygląda rozmiar poziomy eksponowanej całości, i odległości między kolumnami mogą zostać zmienione. Z definicji tej komendy na str. 80 wynika, że domyślny rozmiar eksponowanego materiału jest szerokością strony, a standardowa odległość między kolumnami wynosi 0,2 cala. Nowy rozmiar poziomy (szerokość) jest trochę mniejszy od rozmiaru strony (patrz strona składu). Wartość za znakiem `/` określa

liczbę kolumn.

10: W składzie trzech lub więcej kolumn tekstu na standardowym rozmiarze strony (7 cali lub trochę mniej) może pojawić się problem przepełnionych pudełek (*overfull boxes*), tzn. \TeX nie będzie poprawnie dzielił wierszy. Należy wówczas zmienić parametry `\tolerance` i `\hbadness` (tu została zmieniona wartość `\baselineskip`). Obydwie komendy są omówione w dodatku.

16, 29: Pusty wiersz lub komenda `\par` wskazują koniec akapitu.

39: W tym miejscu kończą się strony gazety. Przywrócone jest wcześniejsze ustawienie `\footline`.

Szafranowe lody

kącik dla kucharzy

Szafran to przyprawa z suszonych kwiatów krokusa, należącego do rodziny irysów. Ma wspaniały zapach i smak, nie posiadające substytutów; to jedna z najdroższych przypraw w świecie — sprzedawana w cenie cztery razy niższej od ceny złota. Mała porcja tej przyprawy wystarcza na długo, dlatego czasem można się na nią skusić.

Dzięki temu przepisowi można zrobić niezapomniane lody. Nawet najmniej zainteresowani jedzeniem przerwą rozmowy, ich oczy rozszerzą się z zaciekawieniem, kiedy

pierwsza łyżeczka lodów dotknie ich języka.

Składniki

- 1/3 filiżanki śmietany.
- 1 duża szczypta włosków szafranu (ok. 0,5 g).
- 12 torebek kardamonu.
- 2,3 l lodów waniliowych.

Wykonanie

Podgrzać śmietanę i dodać szafran. Pozostawić ją na bardzo wolnym ogniu, na godzinę. Jeśli wydaje się, że zaczyna parować, dodać więcej śmietany. Śmietana przybierze intensywnie złoty kolor.

Następnie obrać i rozgnieść nasiona kardamonu, dodać do śmietany, wymieszać wszystko i zdjąć z ognia.

Kilka minut przed podaniem umieścić lody w dużym pucharze i połączyć miksturą ze śmietany. Za pomocą ręcznego miksera ubić śmietanę z lodami. Wymaga to siły fizycznej i dużej uwagi, ale zapłata za wysiłek już jest bardzo blisko. Mieszać należy od brzegu naczynia (łatwiej to robić, jeśli lody nie są bardzo zmrożone).

Podawać od razu.

```

1 \rightline{\title Podziękowania}
2 \rightline{\sl Przełączanie fontów; skład ,,w chorągiewkę'';}
3 \rightline{\sl zakaz podziału wyrazów.}
4 \vskip 2,5 true in \vskip-2\baselineskip
5 \headline{\hfil}
6
7 \font\smrm=plr9          \font\smit=plti9
8 \font\smsl=plsl9        \font\smbf=plbx9
9 \def\medfont {\let\rm=\smrm \let\it=\smit \let\sl=\sml \let\bf=\smbf
10      \baselineskip=12pt \rm}
11
12 \vbox{\hsize 5,6 true in \pretolerance=10000
13 \raggedright \raggedbottom \medfont
14 \noindent Luca Bombelli pomógł mi uczyć się \TeX-a, znosząc cierpliwie wiele
15 naiwnych pytań, jakimi go nękałem, oraz udzielił kilku pożytecznych sugestii.
16 Carl Brown, Harvey Rarback i~Jim Simone bardzo mi pomogli od strony
17 komputerowej. George Greenwade, Rainer Sch\opf i~Philip Taylor pospieszyli
18 z~niemieckim przenoszeniem wyrazów. Aida El-Khadra zaznajomiła mnie
19 z~niemieckim przenoszeniem wyrazów. Kilku innych ludzi popatrzyło na tę
20 książkę z różnych stron, udzieliło mi sugestii, poparcia i zachęty. Byli to
21 Parag Amladi, Abhay Ashtekar, Barbara Beeton, Matt Choptuik, Mike Creutz,
22 Rosanne DiStefano, Josh Goldberg, Isabell Harrity, Nahmin Horwitz, Bob Melter,
23 Barbara Nichols, Frank Paige, Peter Palffy-Muhoray, Tomas Rokicki i~Joette
24 Stefl-Mabry. Dziękuję zwłaszcza Victorowi Eijkhout i~Catherine Wulforst,
25 którzy uważnie przeczytali jedną z~końcowych wersji podręcznika i~zrobili
26 wiele szczegółowych uwag krytycznych. Dziękuję im wszystkim. Również jestem
27 wdzięczny wydawcom z~Academic Press --- Sari Kalin, Camille Pecoul, Natasha
28 Sabath i~Jenifer Swetland --- za entuzjazm, jaki okazali tej książce.
29 \smallskip
30 Szczególnie dziękuję Donaldowi Knuthowi za wiele godzin przyjemności pracy
31 z~\TeX-em, za jego entuzjastyczną i zachęcającą reakcję na pierwszą wersję
32 książki oraz za kilka szczegółowych i~bardzo pomocnych komentarzy.
33 \smallskip
34 Oczywiście ja sam jestem odpowiedzialny za błędy i~inne nieścisłości, które
35 wciąż znajdują się w tekście.
36 \smallskip
37 Pierwsza wersja książki powstała w~1987 roku, kiedy byłem doktorem, wykładowcą
38 w Relativity Group w Uniwersytecie Syracuse, częściowo dzięki grantom NSF
39 PHY8318350 i PHY8310041. Później pracowałem na wydziale uniwersytetu
40 Long Island oraz gościnnie w~High Energy Theory Group w~Brookhaven National
41 Laboratory. Dziękuję moim kolegom z~tych wszystkich instytucji za ich poparcie.
42 \smallskip} \vfil\eject

```

UWAGI

4: Komendy `\vskip` umieszczone są tu w celu uzyskania odstępu 2,5 cala po tytule głównym.

5: Nagłówek, niepotrzebny tutaj, został zlikwidowany.

7–10: Jest to prosta ilustracja zmiany stopnia pisma za pomocą jednej komendy. W wierszach 7–10 zdefiniowano nazwy kilku 9-punktowych fontów z rodziny Computer Modern. `\medfont` pozwala wykorzystywać komendy `\rm`, `\it`, `\sl` i `\bf` dla krojów antykwy, kursywy, pochylego i pogrubionego, również wybiera różne odstępy między wierszami, a `\rm` czyni stylem domyślnym. Komendy takie są przydatne, ponieważ pozwalają wybrać nowy stopień pisma w dokumencie, pozostawiając te same komendy wywołujące odmiany kroju. Odwołania na str. 88 wykorzystują tę możliwość. Aby wrócić do oryginalnych fontów, należy ponownie zdefiniować omówione wyżej komendy. Można także zastosować nawiasy do lokalizacji zasięgu działania komendy `\medfont`.

12: `\pretolerance` określa maksymalny tolerowany odstęp międzywyrazowy przed podziałem słowa. `TeX` przypisuje wartość kary „badness” do zbyt dużych odległości i próbuje aranżować słowa tak, by kara nie przekroczyła wartości `\pretolerance`. W przypadku gdy okazuje się, że kary za odstępy są zbyt duże, program zaczyna dzielić wyrazy, chyba że wartość parametru `\pretolerance` wynosi 10000, co oznacza brak zezwolenia na podział wyrazów. Maksymalna wartość kary przypisanej do odstępu wynosi także 10000.

13: Justowanie pełne wymusza elastyczność odstępow poziomych i pionowych. Odstępy międzywyrazowe i międzyzdaniowe mogą być elastyczne, tak samo jak odstęp między akapitami, czy odstępy `\bigskip`, `\medskip`, itd. Skład w chorągiewkę `\raggedright` ustala sztywne odstępy poziome, `\raggedbottom` zaś — sztywne odstępy pionowe.

Podziękowania

Przełączanie fontów; skład „w chorągiewkę”;

zakaz podziału wyrazów.

Luca Bombelli pomógł mi uczyć się T_EX-a, znosząc cierpliwie wiele naiwnych pytań, jakimi go nękałem, oraz udzielił kilku pożytecznych sugestii. Carl Brown, Harvey Rarback i Jim Simone bardzo mi pomogli od strony komputerowej. George Greenwade, Rainer Schöpf i Philip Taylor pospieszyli z informacją o archiwach T_EX-owych. Aida El-Khadra zaznajomiła mnie z niemieckim przenoszeniem wyrazów. Kilku innych ludzi popatrzyło na tę książkę z różnych stron, udzieliło mi sugestii, poparcia i zachęty. Byli to Parag Amladi, Abhay Ashtekar, Barbara Beeton, Matt Choptuik, Mike Creutz, Rosanne DiStefano, Josh Goldberg, Isabell Harrity, Nahmin Horwitz, Bob Melter, Barbara Nichols, Frank Paige, Peter Palfy-Muhoray, Tomas Rokicki i Joette Stefl-Mabry. Dziękuję zwłaszcza Victorowi Eijkhout i Catherine Wulforst, którzy uważnie przeczytali jedną z końcowych wersji podręcznika i zrobili wiele szczegółowych uwag krytycznych. Dziękuję im wszystkim. Również jestem wdzięczny wydawcom z Academic Press — Sari Kalin, Camille Pecoul, Natasha Sabath i Jenifer Swetland — za entuzjazm, jaki okazali tej książce.

Szczególnie dziękuję Donaldowi Knuthowi za wiele godzin przyjemności pracy z T_EX-em, za jego entuzjastyczną i zachęcającą reakcję na pierwszą wersję książki oraz za kilka szczegółowych i bardzo pomocnych komentarzy.

Oczywiście ja sam jestem odpowiedzialny za błędy i inne nieścisłości, które wciąż znajdują się w tekście.

Pierwsza wersja książki powstała w 1987 roku, kiedy byłem doktorem, wykładowcą w Relativity Group w Uniwersytecie Syracuse, częściowo dzięki grantom NSF PHY8318350 i PHY8310041. Później pracowałem na wydziale uniwersytetu Long Island oraz gościnnie w High Energy Theory Group w Brookhaven National Laboratory. Dziękuję moim kolegom z tych wszystkich instytucji za ich poparcie.

```

1 \line{\hfil\title Cytowana literatura}
2 \line{\hfil\sl Automatyczne numerowanie; listy wewnątrz list.}
3 \headline{\hfil}
4 \vskip 2,5 true in \vskip-2\baselineskip
5
6 \newcount\q \q=0
7 \def\nref {\global\advance\q by1 \item{\bf\the\q.}}
8
9 {\medfont \frenchspacing
10 \nref D.E. Knuth, {\it The \TeX book}, Addison Wesley (1986).
11 \nref G. Orwell, {\sl Politics and English Language}, wznowiony
12 jako \#38 tom 4 w {\it The Collected Essays, Journalism and Letters
13 of George Orwell}, Penguin Books (1970).
14 \nref V.S. Naipaul, {\it A House for Mr.~Biswas}, Andr'e Deutsch (1961);
15 amerykańskie wydanie --- Alfred A.~Knopf (1983). \copyright\ 1961, 1969
16 V.S.~Naipaul.
17 \nref B. Belitt, {\sl The Heraldry of Accommodation: A House for
18 Mr.~Naipaul}, {\it Salmagundi}, \#54, 23--42 (1981).
19 \nref R. Penrose, {\it Rev. Mod. Phys.} {\bf 37}, 215--220 (1965).
20 \nref S.W. Hawkins i G.F.R. Ellis, {\it The Large Scale Structure of
21 Space-time}, Cambridge Univ. Press (1973).
22 \nref V. Moncrief i J. Isenberg,
23 \itemitem{\bf a.}{\it Comm. Math. Phys.} {\bf 89}, 387--413 (1983);
24 \itemitem{\bf b.}{\it J. Math. Phys.} {\bf 26}, 1024--1027 (1985).
25 \nref A. Borde, {\it Phys. Lett.} {\bf A102}, 224--226 (1984).
26 \nref D.E. Knuth, {\it Surreal Numbers}, Addison Wesley (1974).
27 \nref K. G"odel,
28 \itemitem{\bf a.} {\sl A Remark About the Connection Between Relativity
29 Theory and Idealistic Philosophy}, w {\it Albert Einstein:
30 Philosopher-Scientist}, red. P.A. Schilpp, Open Court Publishing Co.
31 (1949);
32 \itemitem{\bf b.}{\sl An Example of a New Type of Cosmological Solution of
33 Einstein's Field Equations of Gravitation}, {\it Rev. Mod. Phys.} {\bf
34 21}, 447--450 (1949).
35 \nref R. Penrose, {\it Phys. Rev. Lett.} {\bf 14}, 57--59 (1965).
36 \nref S.W. Hawkins, {\it Phys. Rev. Lett.} {\bf 15}, 689--690 (1966).
37 \nref H.R. Lewis, {\it Phys. Rev. Lett.} {\bf 18}, 510--512 (1967).
38 \nref F.J. Tipler, {\it Phys. Rev.} {\bf D17}, 2521--2528 (1978).
39 \nref A. Borde, {\it Cl. and Quant. Grav.} {\bf 4}, 343--356 (1987).
40 \nref G.H. Hardy, {\it Ramanujan}, Chelsea Publishing Co. (1978).
41 \nref A. Tomaszewski, {\it Leksykon pism drukarskich},
42 Wyd. Krupski i S-ka (1996).\par}
43 \bye

```

UWAGI

6–7: Nowa zmienna numeryczna `\q` utrzymuje ciąg numeracji odwołań. Przy każdym nowym odwołaniu, rozpoczynającym się komendą `\nref`, `\q` wzrasta o 1, a jego aktualna w danej chwili wartość jest drukowana przez komendę `\the`. Pozwala to na zmianę jednostek w liście bez każdorazowej zmiany numeru.

9: `\medfont` jest komendą włączającą krój pisma zdefiniowany na str. 86. `\frenchspacing` powoduje, że wszystkie pojedyncze odstępy są takie same, niezależnie od miejsca, w którym występują. (Normalnie `\TeX` pozostawia trochę większe odstępy po znakach interpunkcji, np. po kropce.) Ma to znaczenie w przypadku składu bibliografii, czy indeksów. Efekt działania komendy `\frenchspacing` może być albo ograniczony

do grupy (wewnątrz nawiasów), jak to zrobiono w tym miejscu, albo wyłączony później za pomocą komendy `\nofrenchspacing`.

23, 24, itd.: Komenda `\itemitem` składa wyliczenie zagłębione.

42: Kiedy parametry akapitu są lokalnie zmienione (tj. wewnątrz grupy), końcowy akapit musi być zakończony, zanim zostanie zamknięta grupa. Inaczej ten akapit nie będzie honorował nowego ustawienia. Patrz uwagi do str. 6 i 20. Wyrażna komenda końca akapitu `\par` jest wykorzystana tu do zakończenia akapitu przed zamknięciem nawiasu `}`, co pozwala mieć pewność, że nowe ustawienie `\baselineskip` będzie obowiązywać do końca akapitu.

Cytowana literatura

Automatyczne numerowanie; listy wewnątrz list.

1. D.E. Knuth, *The T_EXbook*, Addison Wesley (1986).
2. G. Orwell, *Politics and English Language*, wznowiony jako #38 tom 4 w *The Collected Essays, Journalism and Letters of George Orwell*, Penguin Books (1970).
3. V.S. Naipaul, *A House for Mr. Biswas*, André Deutsch (1961); amerykańskie wydanie — Alfred A. Knopf (1983). © 1961, 1969 V.S. Naipaul.
4. B. Belitt, *The Heraldry of Accommodation: A House for Mr. Naipaul*, *Salmagundi*, #54, 23–42 (1981).
5. R. Penrose, *Rev. Mod. Phys.* **37**, 215–220 (1965).
6. S.W. Hawkins i G.F.R. Ellis, *The Large Scale Structure of Space-time*, Cambridge Univ. Press (1973).
7. V. Moncrief i J. Isenberg,
 - a. *Comm. Math. Phys.* **89**, 387–413 (1983);
 - b. *J. Math. Phys.* **26**, 1024–1027 (1985).
8. A. Borde, *Phys. Lett.* **A102**, 224–226 (1984).
9. D.E. Knuth, *Surreal Numbers*, Addison Wesley (1974).
10. K. Gödel,
 - a. *A Remark About the Connection Between Relativity Theory and Idealistic Philosophy*, w *Albert Einstein: Philosopher-Scientist*, red. P.A. Schilpp, Open Court Publishing Co. (1949);
 - b. *An Example of a New Type of Cosmological Solution of Einstein's Field Equations of Gravitation*, *Rev. Mod. Phys.* **21**, 447–450 (1949).
11. R. Penrose, *Phys. Rev. Lett.* **14**, 57–59 (1965).
12. S.W. Hawkins, *Phys. Rev. Lett.* **15**, 689–690 (1966).
13. H.R. Lewis, *Phys. Rev. Lett.* **18**, 510–512 (1967).
14. F.J. Tipler, *Phys. Rev.* **D17**, 2521–2528 (1978).
15. A. Borde, *Cl. and Quant. Grav.* **4**, 343–356 (1987).
16. G.H. Hardy, *Ramanujan*, Chelsea Publishing Co. (1978).
17. A. Tomaszewski, *Leksykon pism drukarskich*, Wyd. Krupski i S-ka (1996).


```

1 \input tbeappnd
2 \startat{91}{1}
3
4 %UWAGA: ,, "\pn"'' poniżej, określa pogrubiony krój pisma (cmbx12)
5 %      wykorzystany również do numerowania stron.
6 {\let\par=\endgraf \leftskip-.5in \parskip3pt \parindent\indsize
7 \baselineskip13,5pt \start \noindent
8 Dodatek ma układ alfabetyczny. Obejmuje definicje pojęć i~komend.
9 \smallskip
10 \ti {\pn Komendy} można podzielić na dwie grupy: podstawowe ({\it
11 primitive\/}) i~złożone. Podstawowe są głównymi blokami budulcowymi w~TeX-u,
12 nie da się ich rozłożyć na inne komendy. {\emph W~dodatku zostały one
13 oznaczone gwiazdką $\ast$}. Instrukcje złożone są zdefiniowane za pomocą pojęć
14 pierwotnych. Komendy przedstawione w~tym podręczniku są częścią szeroko
15 rozpowszechnionego pakietu {\emph Plain~TeX} (pakiet ten także został
16 omówiony w~dodatku). W~większości przypadków komendy Plain \TeX-a będą
17 traktowane jako podstawowe: zostaną objaśnione za pomocą terminów opisujących
18 efekt, jaki wywołują. W kilku przypadkach zostanie jednak podany sposób ich
19 tworzenia. Pozwoli to zobaczyć (częściowo) wewnętrzną budowę Plain \TeX-a
20 oraz poznać sposób dostosowywania definicji do potrzeb użytkownika.
21 \smallskip
22 \ti {\pn Odszukanie hasła} w~dodatku jest łatwe. Symbole matematyczne można
23 znaleźć pod różnymi podpunktami hasła {\htopic matematyka} (np. {\htopic
24 matematyka:\ alfabet\ grecki}), inne znaki specjalne są wymienione w~opisie
25 {\htopic znaki\ specjalne}, a~komendy dotyczące odstępów --- pod hasłami
26 {\htopic odstępy} i~{\htopic matematyka:\ odstępy}.
27 \smallskip
28 \ti {\pn Ramki} z~plikiem wejściowym nie będą przedstawione w~pozostałej
29 części dodatku, z~wyjątkiem niektórych bardziej skomplikowanych fragmentów,
30 jak np. $\spadesuit$:
31 \beginliteral
32   ... skomplikowanych fragmentów, jak np. $\spadesuit$:
33 \endliteral
34 \noindent Przyzwyczajenie się do takiego ujęcia nie powinno zabrać zbyt
35 wiele czasu.
36 \smallskip
37 Itd.

```

UWAGI

Plik wejściowy dodatku składa się z powtarzających się fragmentów, stąd nie jest konieczne jego wyświetlanie. Tak więc na razie jest to ostatnie pudełko z plikiem wejściowym. Komendy formatujące dodatek (umieszczone w pliku *tbeappnd*) są bardzo interesujące. Zostały przedstawione w uwagach końcowych, tam również znajduje się fragment pliku wejściowego.

W szczególnie interesujących miejscach przedstawiono w składzie oryginalny zapis fragmentów stron parzystych.

Komendy wykorzystane na tej stronie w większości przypadków nie są komendami standardowymi. Ich definicje znajdują się w uwagach końcowych.

Dodatek

Skorowidz i słownik.

Dodatek ma układ alfabetyczny. Obejmuje definicje pojęć i komend.

- **Komendy** można podzielić na dwie grupy: podstawowe (*primitive*) i złożone. Podstawowe są głównymi blokami budulcowymi w T_EX-u, nie da się ich rozłożyć na inne komendy. W dodatku zostały one oznaczone gwiazdką *. Instrukcje złożone są zdefiniowane za pomocą pojęć pierwotnych. Komendy przedstawione w tym podręczniku są częścią szeroko rozpowszechnionego pakietu *Plain T_EX* (pakiet ten także został omówiony w dodatku). W większości przypadków komendy Plain T_EX-a będą traktowane jako podstawowe: zostaną objaśnione za pomocą terminów opisujących efekt, jaki wywołują. W kilku przypadkach zostanie jednak podany sposób ich tworzenia. Pozwoli to zobaczyć (częściowo) wewnętrzną budowę Plain T_EX-a oraz poznać sposób dostosowywania definicji do potrzeb użytkownika.

- **Odszukanie hasła** w dodatku jest łatwe. Symbole matematyczne można znaleźć pod różnymi podpunktami hasła *matematyka* (np. *matematyka: alfabet grecki*), inne znaki specjalne są wymienione w opisie *znaki specjalne*, a komendy dotyczące odstępów — pod hasłami *odstęp* i *matematyka: odstęp*.

- **Ramki** z plikiem wejściowym nie będą przedstawione w pozostałej części dodatku, z wyjątkiem niektórych bardziej skomplikowanych fragmentów, jak np. ♠:

... skomplikowanych fragmentów, jak np. `\spadesuit` :

Przyzwyczajenie się do takiego ujęcia nie powinno zabrać zbyt wiele czasu.

- **Podstawowe możliwości** T_EX-a zostały wyjaśnione w różnych miejscach podręcznika. Alfabetyczny układ nie jest idealny do nauki systematycznej, można jednak uczyć się T_EX-a w sposób zorganizowany. Pojęcia, takie jak pudełka i klej, tłumaczą model leżący u podstaw T_EX-a; akapity, strony oraz związane z nimi wpisy opisują jakościowo mechanizm działania programu. Są to jedynie informacje *ogólne* — szczegóły można znaleźć w książce Donalda Knutha *T_EXbook*.

\advance * 2, 68, 72, 78, 80, 88

Zwiększa wartość licznika określonego przez T_EX-a. Komenda ta jest wykorzystana wewnątrz przez Plain T_EX-a np. do automatycznego powiększania wartości numeru strony przy każdorazowym „wypuszczaniu” strony. Patrz `\output`.

akapity 13, 18–19, 20–23, 58–59

Skład zwykłych akapitów jest trywialny (str. 13). Poniżej przedstawione są dodatkowe informacje dotyczące większej kontroli nad akapitami. Wszystkie wymienione tu komendy są omówione także pod osobnymi wpisami.

- *Wywoływanie wcięcia.* Pierwszy wiersz akapitu jest automatycznie wcinany — jest to standardowe ustawienie, które można zmienić, przez zmianę wartości parametru `\parindent`. Inne wiersze mogą być wcinane za pomocą komendy `\indent`. Bardziej skomplikowane wcięcia można uzyskać za pomocą komend `\hangindent` i `\hangafter`.
- *Zakazanie wcięcia* możliwe jest za pomocą komendy `\noindent`.
- *Ustawianie odstępów.* Do zmiany odstępów między akapitami można wykorzystać komendę `\parskip`, a do zmiany interlinii wewnątrz akapitu — `\baselineskip`.
- *Kształt akapitu:* `\leftskip`, `\looseness`, `\narrower`, `\parshape`, `\prevgraf`, `\rightskip`, `\hangindent`, `\hangafter`.
- *Kontrolowanie końca akapitu* umożliwia komenda `\parfillskip`.
- *Jak T_EX składa akapity?* (`\pretolerance`, `\tolerance`, `\hbadness` i `\hbrief`) T_EX czyta najpierw cały akapit, tak jakby to był jeden długi wiersz (poziomy tryb pracy). Następnie próbuje ciąć go na mniejsze kawałki odpowiedniej długości, korzystając z wyspecyfikowanych wartości `\hfuzz`, `\pretolerance` i `\tolerance` (bierze również pod uwagę inne czynniki, jak kary za niewłaściwy podział wiersza). Stąd zmiana parametru na końcu akapitu ma wpływ na uporządkowanie tekstu na jego początku.
- *Kiedy ustawiać parametry kształtujące akapit?* Wcięcie akapitu jest ustawiane na początku akapitu, jego rozmiar określa wartość `\parindent`. Można zmienić wartość `\parindent` w środku akapitu, jednak nie będzie to miało wpływu na dany akapit, zadziała dopiero w następnym akapicie. Ustawienia `\hangindent`, `\hangafter`, `\parshape` i `\looseness`, wykorzystane przy kształtowaniu akapitu, są komendami działającymi w miejscu, w którym akapit się kończy. Jeśli zostaną użyte w środku akapitu, T_EX wykorzysta wartości ustawione wcześniej. Po złożeniu akapitu, automatycznie przywracane są wartości domyślne (`\hangindent=0pt`, `\hangafter=1`, `\parshape=0`, `\looseness=0`). Wartości `\narrower`, `\leftskip`, `\rightskip` oraz `\baselineskip` i `\parfillskip` również działają na końcu akapitu, jednak po złożeniu akapitu nie są automatycznie przywracane poprzednie wartości; należy ręcznie przywrócić wcześniejsze ustawienia lub użyć nawiasów grupujących do określenia zasięgu działania tych parametrów (w takim przypadku — w odniesieniu do wszystkich parametrów z wyjątkiem `\parindent` — należy zwrócić uwagę, by akapit wyraźnie kończył się przed zamknięciem nawiasu wskazującego koniec grupy).
- *Komendy, które mogą znaleźć się na początku akapitu:* `\indent` i `\noindent`; dowolna komenda nieznakowa (patrz lista komend znakowych); komendy takie jak `\char`, drukujące określony znak; znaki `$` (T_EX rozpoczyna nowy akapit i otwiera tryb matematyczny); komendy tworzące akcenty; komendy dotyczące odstępów poziomych, takie jak `\quad`, `\hskip`, `\hfil` itd.; `\valign`; `\vrule`, `\item` i `\itemitem`. Należy podkreślić, że komenda `\kern` nie znajdzie się na początku akapitu: umieszczona między akapitami jest interpretowana jako komenda odstępów pionowych, a umieszczona wewnątrz właśnie rozpoczętego akapitu jest przez T_EX-a traktowana

jak specyficzny odstęp poziomy. Podobnie ani `\hbox`, ani komendy na niej oparte (jak np. `\line`) nie są traktowane przez T_EX-a jako rozpoczynające akapit.

- *Komendy przerywające akapit:* `\centerline`, `\leftline`, `\rightline` i `\line`; eksponowane wzory matematyczne (umieszczone między podwójnym znakiem \$). Kiedy każda z komend kończy działanie, od następnego wiersza automatycznie kontynuowany jest skład przerwanej akapitu. Należy pamiętać, że niektóre komendy `\line` mogą być użyte dopiero po wyraźnym zakończeniu bieżącego wiersza akapitu (np. za pomocą `\hfil\break`).
- *Komendy kończące akapit:* pusty wiersz; komenda `\par` lub `\endgraf`; odstęp pionowy, jak `\vskip`, `\vfil`, `\smallskip`, itd.; `\item` i `\itemitem` (te kończą dany akapit i rozpoczynają nowy); `\hrule`; `\halign`; komendy kończące stronę, takie jak `\eject`, `\smallbreak`, `\goodbreak` itd.; `\bye`; `\end`.

akcenty (w tekście) [x-xi](#), [36-37](#), [58-59](#)

è \´e	é \’e	ê \^e	ë \" e	ẽ \~e
ē \macron{e}	ė \.e	ě \u e	ě \v e	ě \H e
æ \t ae	ę \c e	ę \d e	ę \b e	

Akcenty te mogą być umieszczone nad dowolną literą. Jeśli akcent ma znaleźć się nad literami „i” lub „j”, najpierw należy je pozbawić kropek (patrz znaki specjalne, przykład IV). Znaku „\t ae” (z tabeli powyżej) nie można złożyć komendą „\tae”. Należy wpisać „\t{ae}” i taki sposób jest częściej preferowany, ponieważ określa obszar działania komendy.

Komenda `\macron` jest charakterystyczna dla spolszczonej wersji Plain T_EX-a, tj. formatu MeX. Zastępuje oryginalną komendę `\=`, wykorzystaną w MeX-u do wstawienia dywizu [przyp. tłum.].

`\allowbreak`

Sugeruje podział wiersza (podział wierszy) w miejscach, których T_EX nie wykorzystał do tego celu. Wyrażenie matematyczne nie będzie złamane w miejscu przecinka, chyba że jest to wyraźnie dozwolone, jak np. w tym miejscu pliku wejściowego: (*a*, *b*, *c*, *d*, *e*, *f*).

```
... w tym miejscu pliku wejściowego:\quad $(a,\allowbreak
b,\allowbreak c,\allowbreak d,\allowbreak e,\allowbreak f)$.
```

argumenty

Argumentem komendy jest najczęściej tekst, na który ona działa. Na przykład argumentem dla `\centerline{Nie zgadzaj się!}` jest `Nie zgadzaj się!`. Nie wszystkie komendy posiadają argumenty i, o ile to jest możliwe, preferuje się stosowanie poleceń bez argumentów. Jeśli komenda przyjmie argumenty, program przeczyta je do końca, zanim wykona komendę. Proces taki pochłania czas i pamięć. W przypadku `\centerline` np. jest to nieuniknione — T_EX szuka, co ma być wyśrodkowane, zanim złoży wiersz. Inna sytuacja występuje przy zmianie kroju pisma. Nowy krój jest włączany w pewnym punkcie i tekst jest składany nim aż do momentu wyłączenia (zmiany kroju). Wówczas program nie musi czytać najpierw dalszego tekstu, aby znaleźć punkt końcowy działania komendy. Można też zdefiniować komendę typu „zmień-na-kursywę” w postaci `\italic{...}`, jednak taka komenda z argumentem będzie wysoce niewydajna. Standardowa procedura składu kursywą `{\it...}` jest dużo lepsza, tzn. zdefiniowanie osobno miejsca początku i końca działania komendy (w przypadku komendy `{\it` punktem końcowym jest miejsce, w którym pojawia się nawias `}`).

at [viii–ix](#)

Słowo kluczowe (*keyword*) wykorzystywane do określenia stopnia kroju pisma.

\baselineskip * [xiv–1](#), [44–45](#), [50–51](#), [58–59](#), [66–67](#), [70–71](#), [74–75](#), [84–87](#)

Reprezentuje wartość odległości między „liniami podstawowymi” kolejnych wierszy. Plain T_EX ustawia ją domyślnie na 12pt. Patrz str. [xiv](#) oraz akapity.

\begingroup *

Komenda grupująca, może być użyta zamiast nawiasu {, by określić początek grupy (w określonych sytuacjach), ale nie zawsze wymiennie z nim działająca. **\begingroup** musi mieć parę w postaci komendy **\endgroup** (patrz **\bgroup** oraz grupowanie).

\bf [vi–vii](#), [2–3](#), [8–9](#), [20–21](#), [86–87](#)

Składa tekst **pogrubionym krojem pisma** (**{\bf pogrubionym krojem...}**).

\bgroup [70–71](#), [80](#)

Komenda traktowana jako nawias otwierający grupę, w wielu sytuacjach może być wykorzystana w miejsce znaku {. Grupa rozpoczęta w ten sposób kończy się zwykle komendą **\egroup**, chociaż w kilku przypadkach możliwe jest użycie nawiasu } (patrz **\let**).

\bgroup i **\egroup** (jak również **\begingroup** i **\endgroup**) pozwalają na większą dowolność w grupowaniu niż nawiasy { i }. Może to mieć znaczenie, gdy istnieje np. konieczność zmiany formatu kilku stron składu. Można wówczas zasignalizować początek zmian za pomocą jednej komendy i wyłączyć ją w dowolnym miejscu inną komendą (patrz **argumenty**). Komendy takie mogą być zdefiniowane za pomocą **\bgroup** (lub **\begingroup**) i **\egroup** (**\endgroup**); zostały użyte w przykładzie XIII (patrz także definicje komend **\ignore** pod hasłem komentarze oraz komend **\table** i **\caption** pod hasłem tabele). Nawiasy { i } nie mogą być wykorzystane w ten sposób. Dodatkowych informacji można szukać pod hasłem grupowanie.

\bigbreak [24–28](#)

Sugeruje T_EX-owi złamanie strony; jeśli złamanie strony nie nastąpi, program składa w tym miejscu odstęp pionowy równy **\bigskip**.

\bigskip [vi–ix](#), [xii–xiii](#), [8–9](#), [11–13](#)

Składa odstęp pionowy: _____ . Jest to odstęp 12-punktowy, który może kurczyć się lub rozciągać, jeśli zachodzi taka potrzeba. Definicja tej komendy w Plain T_EX-u jest następująca:

```
\bigskipamount=12pt plus 4pt minus 4pt
\def\bigskip{\vskip\bigskipamount}
```

W razie potrzeby można w dowolnym miejscu zmienić definicję komendy **\bigskip**; kolejne jej wywołanie będzie działać zgodnie z ostatnią definicją.

błędy

Jeśli w trakcie przetwarzania pliku wystąpi jakikolwiek błąd, program wyświetli od razu odpowiedni komunikat. W niektórych przypadkach sam przeprowadza minimalną „naprawę” uszkodzenia. Nie zmienia to oryginalnego pliku, ale pozwoli T_EX-owi dalej go przetwarzać.

Wiadomość o błędzie może wyglądać następująco:

! *Error message*

A następnie *kilka, zaskakująco wyglądających linii tekstu* (użyte w tekście komendy mogą być zastąpione przez ich definicje za pomocą pojęć podstawowych — czasem nie znanych użytkownikowi — lub może to być odpowiedź programu, jak rozwiązać wynikły problem). Wiadomość o błędzie zawiera także numer i fragment wiersza, w którym błąd wystąpił, a na jej końcu widnieje znak

?

który oznacza, że \TeX czeka na reakcję użytkownika.

A. Reagowanie na wiadomość o błędzie.

W odpowiedzi można korzystać zarówno z dużych, jak i małych liter (<CR> oznacza klawisz Enter):

h<CR> dostarcza dodatkowych informacji o błędzie;

e<CR> lub **x**<CR> zatrzymuje dalsze przetwarzanie pliku, pozwala na powrót do jego edycji; w przypadku poważnych błędów jest to najlepsza opcja;

i<CR> pozwala na wstawienie poprawnego sformułowania, które \TeX może wykorzystać tymczasowo i poprawnie przetwarzać dalej plik; potem należy wyedytować plik, aby nanieść trwałą poprawkę;

<CR> pozwala na dalsze przetwarzanie pliku, po wewnętrznych poprawkach programu; przy niezbyt poważnych błędach wystarczy notować numery wierszy do poprawy i wciskać <CR> w celu dalszego przetwarzania pliku; po przetworzeniu pliku należy go wyedytować i nanieść wszystkie poprawki;

?<CR> wyświetla wszystkie dostępne opcje.

Na następnych stronach omówiono najczęściej spotykane informacje dotyczące błędów.

B. Ogólne klasy wiadomości o błędach.

1. Niepełne i przepełnione pudełko.

Nie są to prawdziwe błędy, gdyż nie zatrzymują pracy \TeX -a, wpływają jedynie na wynik składu. Wiadomości o nich mogą jednak wprawić użytkownika w zakłopotanie, zwłaszcza jeśli jest on nowicjuszem w pracy z \TeX -em. Komunikaty te informują, czy \TeX jest w stanie wypełnić stronę tekstem lub innym materiałem w sposób estetyczny. Jeśli jest za dużo materiału, który należy umieścić na zbyt małej powierzchni, \TeX melduje (*overflow*) przepełnione pudełko, w przeciwnym wypadku — niepełne (*underfull*).

Przepełnione pudełko wygląda fatalnie, zwłaszcza że \TeX , aby zwrócić uwagę na ich wystąpienie, umieszcza na końcu danego wiersza czarny prostokąt. Należy miejsca te przeredagować w pliku wejściowym. Jeśli taka wiadomość dotyczy np. eksponowanego wyrażenia matematycznego, należy podzielić je ręcznie, tak aby znalazło się w dwóch wierszach.

Komunikat *underfull box* oznacza, że w wierszu pojawiły się większe odstępki niż normalnie dopuszczalne (opisy pod hasłami `\hbadness` i `\vbadness` informują, w jaki sposób można zmienić granice tolerancji \TeX -a, dodatkowe informacje można znaleźć także w opisie komendy `\break`). Chociaż komunikaty brzmią alarmująco — „niepełność” jest wyrażona w jednostkach ilościowych zwanych karami (*badness*) — błędy *underfull* zwykle nie mają wpływu na wynik składu i można je zignorować. W niektórych przypadkach wyjściem z sytuacji jest zastosowanie komendy `\hfil` lub `\vfil`, która umieści w odpowiednim miejscu znak odstępu. Plain \TeX wyrównuje górne i dolne marginesy kolejnych stron, stąd w przypadku sztywnych odstęp-

pów pionowych — np. dużej wartości `\baselineskip` — pojawiają się informacje o niepełnych pudełkach. Wówczas pomoc może rozsądne wstawienie elastycznych odstępów (patrz `\parskip`).

2. Brakujący znak \$.

Z reguły łatwo jest znaleźć miejsce, gdzie powinien być brakujący znak \$. `TeX` czasem umieszcza tymczasowo takie znaki, w miejscach, w których powinny one wystąpić. Niekiedy jednak taka wiadomość wprawia użytkownika w zakłopotanie, ponieważ może się mu wydawać, że znak \$ jest w danym miejscu niepotrzebny lub że już go wstawił; z reguły jest tak w przypadku, gdy `TeX` wewnętrznie wykorzystuje tryb matematyczny do definiowania niektórych komend.

Jako przykład można podać komendę `\underbar`, stosowaną do podkreślenia tekstu, która jest zdefiniowana w Plain `TeX`-u za pomocą komendy `\underline`, komendy z trybu matematycznego. Tak więc zastosowanie komendy `\underbar` w trybie matematycznym spowoduje pojawienie się komunikatu o błędzie. Nawet w takich przypadkach znalezienie źródła błędu zwykle nie jest trudne.

3. Brakujący znak { lub }; brak argumentu.

Znalezienie miejsc, w których należy umieścić brakujące nawiasy, może być uciążliwe, zwłaszcza w dużych fragmentach tekstu. Jedną z technik uniknięcia takich pomyłek jest wpisywanie obu symboli (otwierającego i zamykającego) na początku grupy. Wtedy nie trzeba się martwić o to, by pamiętać o zamknięciu wcześniej otwartej grupy. Na przykład, aby wyśrodkować tekst, można wpisać najpierw `\centerline{}`, a następnie umieścić między nawiasami materiał, który ma być wyśrodkowany. Podobnie można postąpić przy zmianie kroju pisma, np. na kursywę; wpisać najpierw `{\it }`, a następnie umieścić w nawiasach tekst, który ma być złożony kursywą.

Najczęściej zapomina się o umieszczeniu nawiasu zamykającego w miejscach, gdzie następowała zmiana kroju pisma lub definiowano nową, skomplikowaną komendę, ponieważ w tych przypadkach grupa zwykle obejmuje kilka wierszy. Źródła takich błędów nie są łatwe do znalezienia, ponieważ wiadomość o błędzie pojawia się trochę później niż błąd lub nawet na końcu pliku, kiedy `TeX` dochodzi do końca pliku i wciąż znajduje się wewnątrz grupy. Program podaje wówczas poziom (*level*) — liczbę wciąż otwartych grup.

Aby zabezpieczyć użytkownika przed tego typu błędami, `TeX` ogranicza działanie niektórych komend (np. `\centerline`) do jednego akapitu. Kiedy program dochodzi do końca akapitu, a nie znalazł punktu oznaczającego koniec działania komendy, przetwarzanie zostaje przerwane i pojawia się informacja, że akapit zakończył się, a grupa nie jest zamknięta.

4. Brakujące liczby.

`TeX` „zaprotestuje”, jeśli nie znajdzie liczby w miejscu, w którym oczekuje wyspecyfikowania pewnej wielkości (np. po komendach `\hskip` czy `\vskip`), i potraktuje brakującą liczbę jako zero. Tak więc `\hskip` bez liczby zostanie potraktowane jako `\hskip0pt`. Takie błędy są łatwe do poprawienia. Jest jednak pewien haczyk: `TeX` czasem będzie się upierał, że brakuje mu liczby, jeśli komenda będzie umieszczona w niewłaściwym miejscu. Dzieje się tak wtedy, gdy po nazwie komendy niebacznie umieszczone zostanie słowo, które `TeX` może zinterpretować jako słowo kluczowe dotyczące tej komendy.

Jako przykład można podać słowo `plus`, które jeśli znajdzie się zaraz za wpisem `\hskip1pt`, zostanie potraktowane jako kontynuacja komendy, słowo kluczowe odnoszące się do elastyczności odstępu poziomego. W takim przypadku należy wpisać komendę `\relax` — tutaj `\hskip1pt\relax plus`. Okazuje się, że `TeX` wyświetli

wiadomość o błędzie, nawet gdy napotka słowo kluczowe na początku następnego wyrazu, np. `\hskip1pt pluszowy`. Wówczas również należy stosować komendę `\relax`. Wykaz wszystkich słów kluczowych znajduje się pod hasłem `słowa kluczowe`.

5. Przekręcenia.

W zależności od kontekstu, \TeX odniesie się do przekręcenia w różny sposób. Niewłaściwie napisana jednostka (powiedzmy `ot` zamiast `pt`) wywoła odpowiedź postaci `illegal unit`; przekręcenie nazwy komendy spowoduje pojawienie się komunikatu `undefined`. W tym ostatnim przypadku można w celu dalszego przetwarzania pliku zastosować korektę: `i\komenda <CR>`, gdzie `komenda` oznacza poprawnie wpisaną komendę. Zmianę tę należy potem umieścić w pliku wejściowym. Najczęściej błąd taki powstaje poprzez sklejeniu się komendy ze słowem po niej następującym.

6. Komenda w złym miejscu i złym czasie.

Części komend nie można wykorzystywać w dowolnym miejscu pliku, np. komendy trybu matematycznego nie mogą być stosowane poza tym trybem pracy programu, większość komend kończących akapit (patrz `akapity`) nie może być użyta wewnątrz obszaru działania komendy `\hbox`, itd. W takim przypadku pojawi się wiadomość o błędzie — nazwa komendy niewłaściwie użytej i tryb, w jakim \TeX znajduje się w momencie jej wywołania. Najczęściej pomyłki dotyczą komend `\raise` i `\lower` oraz `\moveleft` i `\moveright` — pierwsza para działa w poziomym trybie pracy programu, druga w pionowym.

7. Niedostępne fonty.

Błędy związane z fontami to najczęściej zastosowanie niewłaściwej nazwy zewnętrznej fontu lub brak danego fontu w systemie. Należy wówczas dokonać odpowiednich zmian wybranego fontu w pliku.

\botmark *

Patrz `\mark`.

\box *

Komenda `\box` oznacza rejestr, w którym jest zawarta pewna informacja. Dostępnych jest 256 rejestrów typu `box`: `\box0`, ..., `\box255` (patrz także `\count`). Za pomocą wpisu, np. `\setbox0=\hbox{Skrzynka Pandory}`, można umieścić cokolwiek w danym rejestrze. Zawartość rejestru będzie odzyskana za pomocą komend `\box0` lub `\copy0` — obie drukują zawartość pudełka, pierwsza również kasuje rejestr. Na przykład „Skrzynka Pandory, Skrzynka Pandory, Skrzynka Pandory, ” otrzymano z następującego pliku wejściowego:

```
... rejestr. \setbox0=\hbox{Skrzynka Pandory}Na przykład ,, \copy0,
\copy0, \box0, \box0'' otrzymano ...
```

Jak łatwo można zauważyć, druga komenda `\box0` nie działa — pierwsze jej wywołanie już wyczyściło zawartość pudełka. Można również umieścić informację w pudełku pionowym `\vbox` i odzyskać ją w ten sam sposób. `\box0`, ..., `\box9` wykorzystywane są zwykle jako rejestry pomocnicze (nazywane rejestrami *scratch*), a `\box255` jest zarezerwowany jako schowek, w którym umieszczona jest informacja dotycząca numeru strony. Informacje, jak przydzielać rejestry `\box`, można znaleźć w opisie komendy `\newbox`.

W przypadku gdy korzysta się z dużej liczby wprost wprowadzonych pudełek, uzyskuje się bardzo skomplikowaną strukturę strony (pudełka wewnątrz pudełek, wewnątrz pudełek...). \TeX oferuje użytkownikowi komendy „rozpakowujące” zawartość pudełek: `\unhbox` i `\unvbox`, które zachowują się podobnie jak `\box`, z tym

`\break`

że demonstrują najbardziej zewnętrzny poziom spakowania [dopuszczają podział struktury wewnątrz danego pudełka; szczegółów należy szukać w literaturze, np. *TeXbook* — przyp. tłum.]; `\unhcopy` i `\unvcopy` są analogiczne do `\copy`.

`\break` x-xii, 18–19, 54–56

W zależności od kontekstu, dokonuje podziału wiersza lub łamie stronę. Jeśli \TeX znajduje się w poziomym trybie pracy (jest w środku wiersza; patrz tryby pracy), komenda `\break` powoduje podział wiersza; jeśli \TeX jest w trybie pionowym (między wierszami), wymusza złamanie strony. W przypadku gdy komenda ta zostanie wywołana, a wiersz lub strona nie będą wypełnione justunkiem za pomocą komend `\hfil` czy `\vfil`, \TeX wyświetli informację o błędzie (*underfull box*).

`\buildrel`

Patrz matematyka: efekty specjalne.

`by`

Patrz słowa kluczowe.

`\bye` xii, 17, 58, 88

Komenda złożona, kończąca sesję \TeX -a: kończy akapit przez umieszczenie znaku końca akapitu `\par`, wypełnia pozostałą część strony justunkiem i sprawdza, czy wstawki w rodzaju `\midinsert`, `\topinsert` lub `\pageinsert` zostały uwzględnione przez program. `\bye` jest komendą częściej wykorzystywaną jako znak końca pliku niż `\end`, należy jednak uważnie ją stosować, bo w przypadku gdy przedefiniowana została komenda `\par`, np. w celu wstawiania specjalnych znaków na końcu akapitu, komenda `\bye` może spowodować zbyt wiele ciekawych efektów końcowych.

`\cal` 14–15, 30–35

Składa tekst Kaligrafowanym Krojem Pisma. Komenda ta działa tylko w trybie matematycznym i dotyczy wyłącznie dużych liter (np. `\cal X` składa \mathcal{X}). Dostępne litery w tym kroju to:

$$\begin{array}{cccccccccccccc} A & B & C & D & E & F & G & H & I & J & K & L & M \\ N & O & P & Q & R & S & T & U & V & W & X & Y & Z \end{array}$$

Zastosowanie tej komendy w odniesieniu do małych liter daje zupełnie inne znaki: np. `\cal x` składa znak \mathfrak{x} . Dlatego w przypadku wykorzystywania stylu kaligrafowanego w dłuższych wyrażeniach, należy dokładnie grupować części tekstu, które mają być tak złożone.

`całki`

Patrz przykład VII.

`\cases` 26–27

Składa wyrażenia eksponowane postaci:

$$|x| = \begin{cases} +x & \text{dla } x \geq 0; \\ -x & \text{dla } x < 0. \end{cases}$$

(patrz także przykład II). Plik wejściowy powyższego fragmentu ma postać:

$$|x| = \begin{cases} x & \text{dla } x \geq 0 \\ -x & \text{dla } x < 0 \end{cases}$$
\catcode *

Patrz kategorie znaków.

\centerline vi–vii, 6–9, 20–21, 44–45

Wyśrodkowuje wyrażenie;

np. plik wejściowy tego fragmentu ma postać:

```
\centerline{Wyśrodkowuje wyrażenie;}
```

Kiedy wykorzystujemy tę komendę wewnątrz akapitu, poprzedni wiersz musi być wyraźnie zakończony (np. `\hfil\break`). Patrz także komenda `\line`. Komendy takie jak `\centerline`, oparte na komendzie `\line`, umieszczają materiał w odniesieniu do rozmiaru poziomego `\hsize`: ignorują komendy dotyczące kształtu akapitu, jak `\narrower`, `\rightskip` czy `\leftskip`.

\char *

Patrz kody znaków.

\columns 44–45Określa liczbę kolumn tabel składanych za pomocą komendy `\settabs` (patrz tabela 1 w przykładzie VI oraz `tabulowanie`).**\copy** *Patrz `\box`.**\count** * 68

TeX umieszcza wartości liczb całkowitych w jednym z 256 rejestrów, oznaczonych `\count0`, ... `\count255`. Na przykład numery stron zwykle są umieszczone w rejestrze `\count0` (który ma drugą nazwę `\pageno`). Można umieścić tu swoje własne wartości (jak to zademonstrowano w przykładzie XII). `\count255` jest zwykle zarezerwowany do tymczasowego wykorzystania jako rejestr pomocniczy (*scratch*). Na przykład `\count255=30` umieszcza liczbę 30 w `\count255`, a `\the\count255` podaje wartość rejestru (liczbę 30). Plik wejściowy ostatnich wierszy ma postać:

```
a "\the\count255" podaje wartość rejestru (liczbę
\the\count255). Plik...
```

Umieszczenie liczby w danym rejestrze kasuje jego wcześniejszą zawartość, stąd należy uważnie zmieniać zawartości rejestrów, gdyż niepożądane byłoby usunięcie zawartości rejestru `\count0` czy dziewięciu kolejnych rejestrów (`\count1`, ..., `\count9`), w których są zwykle umieszczone specjalne opisy strony. Dyskusję na temat przydzielania rejestrów można znaleźć pod hasłem `\newcount`. Przydział rejestrów jest lokalny, tzn. zmienna zawarta w danym rejestrze wewnątrz grupy nie wpływa na zawartość rejestrów poza tą grupą (patrz `grupowanie`). Przykładowo, plik wejściowy ostatnich kilku wierszy ma postać:

```
Przykładowo, {\pageno=1000}
%SPRAWDZAMY CZY PROGRAM PRZYJĄŁ WIADOMOŚĆ O ZMIANIE NUMERU STRONY
plik wejściowy...
```

Po przetworzeniu pliku numeracja stron dalej jest poprawna.

\cr

\cr * 26–27, 38, 40–45, 48–51, 54–55, 66–67

Carriage return („powrót karetki”) oznacza koniec wiersza w wyrównaniu typu tabela czy macierz. Funkcjonuje także powiązana z nią komenda podstawowa `\crr`, która działa podobnie jak `\cr`, z wyjątkiem sytuacji, kiedy występuje po komendach `\cr` lub `\noalign` — wtedy nie działa. Komenda ta jest przydatna przy definiowaniu makr wyrównań, w miejscach, w których użytkownik nie umieścił komendy `\cr` [szczegóły można znaleźć w literaturze, np. *T_EXbook* — przyp. tłum.].

\day *

T_EX „rozmawia” z komputerem na początku każdej sesji pracy i „pyta go” o datę; np. dzisiaj jest 17. dzień miesiąca. Plik wejściowy w tym przypadku ma postać:

```
... o~datę; np. dzisiaj jest \number\day. dzień miesiąca.
```

Można także uzyskać informacje dotyczące miesiąca i roku (patrz `\month` i `\year`).

\def * 2, 66, 68, 70, 72, 76, 78, 80, 86

Pozwala definiować komendy użytkownika (patrz przykład XI, uwagi końcowe, definicje i nowe komendy w podręczniku).

definicje 2, 66, 68, 70, 72, 76, 78, 80, 86

Komendy podstawowe `\def` i `\let` pozwalają definiować własne polecenia. Użytkownik, który pozna lepiej program, będzie mógł stwierdzić, jakie to udogodnienie. Będzie mógł zdefiniować swój własny format, w kilku komendach (mogą to być również komendy znakowe) umieścić zestawy instrukcji szczegółowych. Takie komendy formatujące były wykorzystane przy formatowaniu dodatku (patrz uwagi końcowe).

\dimen *

Działa podobnie jak `\count`, z tym że rejestry `\dimen` zawierają zmienne, które posiadają jednostki (wymiar); np. zmienne reprezentujące rozmiar (jak `\hsize`) lub określające wcięcie akapitu. `\dimen0`, ..., `\dimen9` i `\dimen255` są rejestrami pomocniczymi *scratch* (patrz `\newdimen`).

 \displaylines 60–63

Umożliwia eksponowanie listy wyśrodkowanych równań (patrz przykład IX). Komenda `\displaylines` czasem jest najlepszym sposobem na eksponowanie równania, które ma być podzielone na kilka wierszy (patrz matematyka: podział wierszy).

 \displaystyle * 62–63

Wywołuje odstępy, kroje pisma itd., charakterystyczne dla eksponowanego trybu matematycznego (wykorzystywane automatycznie po umieszczeniu wyrażenia wewnątrz podwójnego znaku $\$$). Za pomocą tej komendy styl eksponowany może być wykorzystany bez eksponowania wyrażenia (patrz matematyka: wyrażenia eksponowane).

\divide * 78–79, 80–81

Pozwala dokonywać operacji dzielenia liczników umieszczonych w rejestrach T_EX-a przez liczby całkowite.

\dp *

Określa głębokość (odsadkę) pudełek. Na przykład przy składzie strony, materiał na niej znajdujący się \TeX umieszcza w pudełku `\box255`; po wypuszczeniu strony pudełko to jest puste. Głębokość takiego pudełka można uzyskać np. za pomocą wpisu `\dp255`. Wartość ta może zostać wydrukowana komendą `\the\dp255`. W tym miejscu głębokość pudełka `\box255` wynosi 0.0pt, co oznacza, że jest ono puste (patrz także `\ht`).

dvi

Skrót od *DeVice Independent* (niezależny od urządzenia). Kiedy \TeX przetwarza plik, np. *nazwa-pliku.tex*, przenosi kodowane instrukcje do nowego pliku o nazwie *nazwa pliku.dvi*. Plik ten zawiera opis każdej ze stron, specyfikację użytego kroju pisma, położenie wszystkich znaków, itd., przedstawione w sposób niezależny od drukarki. To właśnie plik *.dvi* jest czytany przez *drivery* drukarki czy sterownik monitora i następnie konwertowany na podgląd.

Chociaż tworzenie pliku przejściowego może się wydawać niepotrzebną komplikacją, jest jednak bardzo przydatne, gdyż raz utworzony (ten sam) plik *.dvi* może być czytany przez różne urządzenia (stąd nazwa „niezależny od urządzenia”) i końcowy dokument będzie wyglądał (w większości przypadków) tak samo na każdym z urządzeń [różnice mogą wynikać ze sposobu wczytania pliku *.dvi*, np. niektóre *viewer-y* nie czytają grafiki postscriptowej — przyp. tłum.]. Tak więc strona oglądana na ekranie, wydrukowana na drukarce laserowej czy na profesjonalnej maszynie drukarskiej, będzie zawsze mieć taką samą zawartość, położenie. Wszystkie te urządzenia produkują cyfrowy obraz (złożony z pojedynczych kropek), a różnice w końcowym wyglądzie dokumentu zależą jedynie od ich rozdzielczości.

Obecnie standardowa rozdzielczość wynosi około 70 punktów na cal w przypadku większości monitorów, 300–600 (a nawet 1200) punktów na cal w przypadku drukarek laserowych oraz 1200–5400 lub więcej punktów na cal w przypadku naświetlarek. Porównując wydruki pochodzące z drukarki laserowej i z naświetlarki, widzimy różnicę, dotyczy to zwłaszcza małych fontów (10-punktowych lub mniejszych), które na drukarce wychodzą bardziej ściśnięte niż z naświetlarki. (Dlatego często wydruki próbne i inne dokumenty drukowane na drukarce laserowej są zwykle składane fontem 12-punktowym lub powiększane za pomocą `\magstep1`.) Przy tworzeniu książek takich jak ta, pliki przetwarzane w \TeX -u są podglądane na ekranie, próbne wydruki powstają na drukarce laserowej i kiedy wszystko jest już w porządku, plik *.dvi* przetwarza się na plik PostScriptowy i naświetla.

Pliki *.dvi* mogą być przesyłane elektronicznie, ale należy uważać, by nie została zniekształcona zawartość pliku. Lepiej jest jednak przysyłać oryginalny plik *.tex*. Pliki *.dvi* mogą być bez problemów przegrywane na kasety czy dyski.

dzielenie wyrazów 22–23, 36–37, 86–87

\TeX sam znajduje miejsca podziału słów. Aby wyraz nie był podzielony, należy umieścić go w pudełku `\hbox`. Możliwe miejsca podziału wyrazów można zasu-gerować programowi za pomocą znaków `\-`. Jest to tzw. opcjonalne dzielenie (łącznik dyskrecjonarny). Wiersze powyżej zawierają dwa takie łączniki w miejscach, których \TeX sam nie wybrał w tym celu. Plik wejściowy tych wierszy ma postać:

```
\TeX\ sam znajduje miejsca podziału słów. Aby
wyraz nie był podzielony, nale\~zy umieścić go w pudełku "\hbox".
Możliwe miejsca podziału wyrazów można zas\~gerować programowi
za pomocą znaków "\-"...
```

Można zauważyć, że \TeX ignoruje niektóre sugestie, a przyjmuje inne. Uciążliwym byłoby umieszczanie za każdym razem łączników dyskrecjonarnych w często pojawiającym się wyrazie. Możliwe jest zrobienie sugestii globalnej, za pomocą komendy podstawowej `\hyphenation{}`, np.

```
\hyphenation{za-su-ge-ro-wać łącz-nik}
```

Wyrazy na liście muszą być oddzielone spacjami. Każdorazowo po napotkaniu w tekście danego słowa \TeX będzie brał pod uwagę sugestie dotyczące miejsc podziału. Należy tu jeszcze omówić komendę `\discretionary`, która pozwala nie tylko sugerować miejsca podziału słów, ale również określa, jak należy sylabizować fragmenty słowa przed i za miejscem podziału oraz jak sylabizuje się pełne słowo (nie dzielone). Jest to pożyteczne w językach, w których podział może zmienić ortografię słowa, lub przy sugerowaniu miejsc podziału w środku ligatur. Na przykład słowo niemieckie *backen* można dzielić wyłącznie jako bak-ken. W \TeX -u należy to wpisać następująco:







```
ba\discretionary{k-}{k}{ck}en
```

Jeśli słowo jest często wykorzystywane, lepszym wyjściem będzie zdefiniowanie nowej komendy, z wbudowaną komendą `\discretionary`.

Wersje \TeX -a od 1990 roku (wersja 3.0 i dalsze) mogą mieć równocześnie wbudowane miejsca podziału z kilku różnych języków [maksymalnie 255 — przyp. tłum.].

efekty specjalne

Umożliwiają wypełnienie odstępu poziomego dowolnego typu wzorem:

	<code>\hrulefill</code>
	<code>\dotfill</code>
	<code>\leftarrowfill</code>
	<code>\rightarrowfill</code>
	<code>\upbracefill</code>
	<code>\downbracefill</code>

Komenda `\dotfill` została wykorzystana w spisie treści. Inne wzory służące do wypełniania odstępów można uzyskać za pomocą komendy `\leaders`.

`\egroup` 70–71, 80

Wykorzystana zwykle do zamknięcia grupy otwartej za pomocą komendy `\bgroup`.

`\eject` vi, viii, 8

Kończy stronę. Patrz łamanie stron.

eksponowane wyrażenia matematyczne

Patrz matematyka: wyrażenia eksponowane.

`\else` * 70, 72

Wykorzystana w połączeniu z `\if...` w komendach warunkowych.

em, ex

Są to zależne od kroju pisma jednostki miary, przydatne w przypadku gdy rozmiar ma się zmieniać automatycznie wraz ze zmianą stopnia kroju pisma. Pojęcia te są terminami drukarskimi. Historycznie *em* było zdefiniowane jako szerokość litery pochylej *M*, w T_EX-u jest to jednostka zależna od kroju pisma. Na przykład odległości między nawiasami [] i [] są równe 1 em, z tym że druga odnosi się do kroju pogrubionego. Odstępy te złożone zostały za pomocą komend `[{\hskip1em}]` i `[{\bf\hskip1em}]` (patrz także `\quad`). Jednostka *ex* jest wysokością litery *x* w obowiązującym w danej chwili kroju pisma.

\end * vi

Określa koniec pliku w T_EX-u. W wielu sytuacjach komenda `\bye` jest jednak lepszym sposobem zakończenia pliku.

\endgraf 70–71

Zdefiniowana w Plain T_EX-u komenda, często używana zamiast komendy `\par`: `\let\endgraf=\par`. Jest przydatna w sytuacjach, gdy komenda `\par` została przedefiniowana, tak jak np. w dodatku (patrz uwagi końcowe). Inny przykład takiego przedefiniowania występuje w definicji komendy `\beginscript` w przykładzie XIII.

\endgroup *

Zamyka grupę otwartą za pomocą komendy `\begingroup`.

\enskip

Składa taki[] odstęp, zdefiniowany następująco:

```
\def\enskip{\hskip,5em\relax}
```

\enspace

Składa taki[] odstęp, zdefiniowany przez:

```
\def\enspace{\kern,5em }
```

Odstęp ten jest „niełamliwy” (wiersz nie może być w tym miejscu podzielony), w przeciwieństwie do odstępu tworzonych za pomocą komendy `\enskip`.

\eqalign\$ 40–43

Wyrównuje równania (patrz przykład V).

\eqalignno\$ 38–41, 56–57

Wyrównuje równania, równocześnie je numerując (patrz przykłady V i VII).

\eqno\$ * 26–29, 40–43

Numeruje równania, umieszczając numery z prawej strony równań (patrz równania 1–3 w przykładzie II lub równanie 4 w przykładzie V).

\every... *

Klasa komend obejmująca: `\everycr`, `\everydisplay`, `\everyhbox`, `\everyjob`, `\everymath`, `\everypar` i `\everybox`. Zasadę działania tego typu komend można pokazać na przykładzie komendy `\everymath`, która umieszcza dany element (tekst) za każdym razem po włączeniu trybu matematycznego. Na przykład fragment „Zawsze matematyka! $1 + 1 = 2$ ” i „Zawsze matematyka! $\int \alpha^2 d\alpha$ ” powstał z pliku wejściowego:

\fi

```
... Na przykład fragment\hfil\break
{\everymath={\hbox{\rm Zawsze matematyka!\ }}}
,,$1+i=2$'' i ,,$\int\alpha^2d\alpha$''.\hfil\break
powstał z pliku...
```

Nawiasy wykorzystano w celu ograniczenia obszaru działania `\everymath`.

Pozostałe klasy tych komend mogą być wykorzystane w ten sam sposób do umieszczenia dowolnego elementu po każdym `\cr`, na początku każdego eksponowanego wyrażenia matematycznego, czy każdego pudełka `\hbox`, na początku akapitu, itd. Nie ma komend typu `\everypage`; do wstawienia dowolnego elementu na każdej stronie wykorzystuje się komendy `\headline` i `\footnote`. Normalnie `\headline` obejmuje jeden wiersz tekstu, jednak są sposoby na umieszczenie kilku linii tekstu w nagłówku. Na przykład numery stron dodatku i wpisy są umieszczone za pomocą komendy `\headline`, ale kreska pozioma już nie (odpowiednie komendy przedstawiono w uwagach końcowych).

`\fi` * 70–73, 80–81

Kończy komendy warunkowe (patrz `\ifdim`).

`\filbreak`

Sugeruje T_EX-owi złamanie strony (patrz łamanie stron).

`\firstmark` *

Patrz `\mark`.

`\folio`

Komenda drukująca numer strony wprowadzony do pamięci komendą `\pageno`. Jeśli przechowywany numer strony (numerowanie stron) jest ujemny, program konwertuje go na cyfrę rzymską właśnie za pomocą komendy `\folio`. Definicja tej komendy w Plain T_EX-u jest następująca:

```
\def\folio{\ifnum\pageno<0 \romannumeral-\pageno
\else\number\pageno \fi}
```

`\font` * viii–ix, vix–1, 64–65, 80–81

Definiuje komendę wywołującą fonty (patrz przykład X).

fonty viii–ix, xiv–1, 8–9, 14–15, 64–65, 80–81

[Font to zapisany w postaci cyfrowej obraz kroju pisma, czyli zestaw informacji o kształtach poszczególnych liter i znaków danego kroju (A. Tomaszewski, *Leksykon pism drukarskich*, 1996) — przyp. tłum.]. Font jest kompletnym asortymentem pisma jednego kroju. Dany krój może występować w kilku odmianach: np. antykwa może być stosowana jako odmiana *pochyła*, *kursywa*, czy **pogrubiona**. Te wariacje są czasem odrębnymi fontami.

Każdy dobry program do składu tekstu potrzebuje dobrych fontów. Równocześnie z T_EX-em Donald Knuth stworzył METAFONT, program do tworzenia fontów. Za pomocą tego programu oraz dzięki wsparciu kilku osób (H. Zapf, M. Carter, R. Southall, N. N. Bilawala, Ch. Bigelow i K. Holmes) stworzył rodzinę fontów *Computer Modern*.

Szesnaście z nich jest ładowanych automatycznie z Plain T_EX-em; są one wywoływane za pomocą standardowych, wcześniej zdefiniowanych komend (fonty te zostały wymienione poniżej). Inne fonty z tej rodziny są również powszechnie dostępne, ale należy uzyskać dostęp do nich za pomocą komendy `\font`. Komenda ta łączy zewnętrzną nazwę fontu (tj. nazwę pliku, w którym jest umieszczona informacja dotycząca rozmiaru znaków w foncie) z nazwą nadaną przez użytkownika. Za pomocą tej nazwy można wywoływać font (patrz przykład X).

Poniżej wymienione są powszechnie dostępne fonty Computer Modern. Fonty już załadowane oznaczone są gwiazdką (*). Nazwa po lewej jest zewnętrzną nazwą fontu: `pl` oznacza spolszczoną wersję fontu Computer Modern [w oryginale były to fonty `cm` — przyp. tłum], a liczba określa rozmiar fontu w punktach (1 cal = 72,27 punktu).

- * `plr10`: antykwa; 10 punktów.
- * `plsl10`: *pochyły*; 10 punktów.
- * `plti10`: *kursywa tekstowa*; 10 punktów.
- `plu10`: wyprostowana kursywa; 10 punktów.
- * `plmi10`: *kursywamatematyczna*, 10pt
- `plmib10`: ***pogrubionakursywamatematyczna***, 10pt
- * `plsy10`: $\mathcal{M} \cup \langle \mathcal{S} \updownarrow \lfloor \updownarrow \infty \prime \sqrt{\cap} \parallel \sqcup \exists \sphericalangle \checkmark \rangle$
— symbole matematyczne
- * `plbx10`: **półgruby rozszerzony**; 10 punktów.
- `plb10`: **półgruby nierozszerzony**; 10 punktów.
- `plbxsl10`: ***półgruby rozszerzony, pochyły***; 10 punktów.
- `plbxti10`: ***półgruby rozszerzony, kursywa***; 10 punktów.
- * `pltt10`: maszynowy; 10 punktów.
- `plitt10`: *kursywa maszynowa*; 10 punktów.
- `plcsc10`: KAPITALIKI; 10 PUNKTÓW.
- `pldunh10`: dunhil; 10 punktów.
- `plff10`: funny; 10 punktów.
- `plfib8`: fibonacci; 8 punktów.
- `plss10`: bezszeryfowy; 10 punktów.
- `plssbx10`: **bezszerfowy, półgruby rozszerzony**; 10 punktów.
- `plssdc10`: **bezszerfowy, półgruby wąski**; 10 punktów.
- `plssi10`: *bezszerfowy, kursywa*; 10 punktów.
- `plssq8`: bezszeryfowy, wyróżniony; 8 punktów.
- `plssqi8`: *bezszerfowy, wyróżniony, kursywa*; 8 punktów.

Oto kilka przykładowych wierszy pliku wejściowego, z którego powstała ta lista:

```
\medskip
{\baselineskip=13pt \parindent1,6in \parskip1pt
\item{\ast$ plr10:} \quad antykwa; 10 punktów.
\item{\ast$ plsl10:} \quad {\sl pochyły; 10 punktów.}
\item{\ast$ plti10:} \quad {\it kursywa tekstowa; 10 punktów.}
\item{plu10:} \quad \font\temp=plu10 {\temp wyprostowana kursywa;
10 punktów.}
\item{\ast$ plmi10:} \quad \font\temp=plmi10 {\temp kursywa
matematyczna; 10 pt.}
\item{plmib10:} \quad \font\temp=plmib10 {\temp pogrubiona kursywa
```



```

matematyczna; 10 pt.}
\item{ $\ast$  plsy10:}\quad \font\temp=plsy10 {\temp Math Symbols;
10 punktów.}
...
\item{plssqi8:}\quad \font\temp=plssqi8 {\temp bezszeryfowy,
wyróżniony, kursywa; 8 punktów.}
\smallskip}

```

W tym pliku wejściowym tymczasowa komenda `\temp` jest przypisana po kolei każdemu z fontów i jako taka wykorzystywana do wywołania fontu.

Spośród standardowych fontów (oznaczonych gwiazdką), czwarty to prawidłowa kursywa matematyczna, która jest włączana (wyłączana) automatycznie, gdy program napotka znak `$`. Kolejny (piąty) jest krojem symboli matematycznych, uruchamianym w taki sam sposób. Komendą `\cal` można uzyskać inne znaki — kaligrafowane (tylko duże litery). Inne standardowe odmiany pisma z tej listy mogą być uzyskane za pomocą zwykłych komend `\rm`, `\sl`, `\it`, `\bf` i `\tt`. `\rm` jest domyślnym krojem pisma w Plain TeX-u. Należy podkreślić fakt, że różne fonty kierują się różnymi zasadami dotyczącymi odstępów i mogą w różny niezwykły sposób reprezentować standardowe znaki (np. fonty matematyczne).

Przedstawione odmiany są dostępne w większości wersji TeX-a, nieznacznie mogą się różnić w różnych systemach. W trakcie ładowania fontu TeX szuka informacji o nim w pliku o nazwie typu *nazwa-zewnętrzna.tfm* (*TeX font metric*); jeśli nie może go znaleźć, informuje o tym. Użytkownik TeX-a ma dostęp do fontów z alfabetów innych niż angielski lub też do innych fontów, nie będących standardowym alfabetem. Oto przykład fontu o nazwie *lcircle 10*:



Plik wejściowy:

```

\font\temp=lcircle10
\centerline{\obeyspaces\temp
a b c d e f g h i j k l m n o
p q r s t u v w x y z}
\bigskip

```

Takie fonty mogą być wykorzystywane na rysunkach, inne, zwane fontami *halftone*, stosowane do reprodukcji fotografii.

Wiele krojów pisma występuje w pewnym zakresie stopni: np. font *plr* zwykle występuje w rozmiarach 5, 6, 7, 8, 9, 10, 12 i 17 punktów, z których 5- i 7-punktowy są bezpośrednio osiągalne. Fonty o tych rozmiarach są również bezpośrednio dostępne w innych odmianach; odmiana *script size* w kursywie matematycznej, w rzeczywistości ma 7-punktowy rozmiar.

Można powiększać rozmiary za pomocą komendy `scaled` lub specyfikować rozmiar za pomocą `at` (patrz przykład X oraz powiększanie). Drukarka może nie uporać się z wszystkimi rozmiarami i powiększeniami, tak więc należy czasem trochę poeksperymentować.

Chociaż fonty można powiększać, w rzeczywistości lepiej wykorzystywać je w rozmiarze, w jakim zostały zaprojektowane (stworzone). Oto przykład różnicy pomiędzy 17-punktową antykwa a 10-punktową, powiększoną do 17 (dokładnie 17,28) punktów:

antykwa 17pt antykwa 10pt, powiększona

A oto lista wszystkich fontów ładowanych w T_EX-u; wymieniona została opisowa nazwa zewnętrzna oraz komenda w T_EX-u, która wywołuje dany font:

plr10	<code>\tenrm</code>	plr7	<code>\sevenrm</code>	plr5	<code>\fiverm</code>
plmi10	<code>\teni</code>	plmi7	<code>\seveni</code>	plmi5	<code>\fivei</code>
palsy10	<code>\tensy</code>	palsy7	<code>\sevensy</code>	palsy5	<code>\fivesy</code>
plbx10	<code>\tenbf</code>	plbx7	<code>\sevenbf</code>	plbx5	<code>\fivebf</code>
plsl10	<code>\tensl</code>	plsl7	<code>\sevensl</code>	plsl5	<code>\fivesl</code>
plex10	<code>\tenex</code>				

Umownie zarezerwowano standardowe nazwy `\rm`, `\sl`, `\it`, `\bf` itd. dla antykiwy, pochylego pisma, itd., niezależnie od stopnia danego kroju pisma. Ponieważ standardowym stopniem jest 10 punktów, `\tenrm` i `\rm`, `\tensl` i `\sl`, itd., zwykle oznaczają to samo. Po włączeniu np. rozmiaru fontu 7 punktów, `\rm` umownie oznacza `\sevenrm`, itd. Widać to w pewnych sytuacjach, np.

$$(\text{wzór})^{\text{potęga}}$$

otrzymano z wpisu:

```
$$({\rm wzór})^{\rm potęga}$$
```

Komenda `\rm` zatem daje pismo różnego stopnia, w zależności od miejsca jej zastosowania. Problem rozmiaru kroju pisma jest omówiony także w uwagach do podziękowań.

\footline 4, 80–81, 84–85

Składa stopkę. Plain T_EX automatycznie umieszcza w niej numer strony (patrz poniżej). Można zmienić to ustawienie za pomocą komendy `\footline` (tak jak zmian w nagłówku dokonuje się wykorzystując komendę `\headline`). Przy definiowaniu wartości `\footline` dobrze jest wyraźnie określić krój pisma, inaczej T_EX zastosuje w stopce krój, jaki wykorzystywał w momencie przechodzenia z trybu składu strony do składu stopki. Patrz uwagi na str. 4.

Definicja komendy `\footline` jest następująca:

```
\footline={\hss\tenrm\folio\hss}
```

Komendy `\hss` i `\folio` są omówione oddzielnie; `\tenrm` oznacza 10-punktową antykwę (patrz także fonty).

\footnote 24–27, 34–35, 44–45

Składa przypis (patrz przykłady II, IV i VI oraz `\vfootnote`). W przykładach zademonstrowany został domyślny styl przypisu w Plain T_EX-u, który można łatwo zmieniać. Można np. zastosować mniejszy krój pisma, można zdefiniować komendę włączającą font mniejszego stopnia (patrz podziękowania i uwagi końcowe) i wykorzystać ten rozmiar w tekście przypisu. Można także modyfikować kreskę (`rule`) oddzielającą przypis od tekstu. Domyślną kreskę określa komenda zwana `\footnoterule`, której definicja w Plain T_EX-u jest następująca:

```
\def\footnoterule{\kern-3pt \hrule width 2 true in \kern2,6pt}
```

Przez skopiowanie tej definicji do pliku, wyedytowanie i dokonanie zmian można zmienić parametry kreski. I na koniec, jednym ze sposobów zmiany wcięcia przypisu jest zmiana wartości `\parindent` bezpośrednio przed użyciem komendy `\footnote` (i później jej przywrócenie).

frakcje górna i dolna 24–29, 36–44, 52–57

Patrz indeksy oraz matematyka: odstępy.

\frenchspacing 88–89

Powoduje, że wszystkie spacje mają równą szerokość, nawet te po znakach interpunkcji. Cecha ta jest pożyteczna przy składzie bibliografii czy innego tekstu, zawierającego wiele skrótów. Do normalnych reguł odstępowych można wrócić za pomocą komendy **\nofrenchspacing** lub przez ograniczenie efektu działania komendy **\frenchspacing** do danej grupy (np. ujęcie materiału w nawiasy **{ }**).

\global * 72–73, 80–81, 88–89

Pozwala na globalne przypisanie wartości zmiennym, zdefiniowanie komend, itd. (tzn. na stworzenie oznaczeń, które wykraczają poza określoną, konkretną grupę, w której się znajdują). Patrz przykład XIV.

głębokość (depth) 12–13, 48–49, 76–85

Każde pudełko stworzone przez T_EX-a powinno mieć linię podstawową (patrz pudełko). Głębokość (depth) odnosi się do pionowego rozmiaru pudełka, poniżej tej linii. Patrz **\dp**. Słowo **depth** to słowo kluczowe przy specyfikowaniu rozmiaru kresek.

\goodbreak

Sugestia dla T_EX-a, słabsza niż **\break**, dotycząca miejsca złamania strony (patrz łamanie stron).

grafika

Patrz wykresy.

grupowanie

Pojęcie grupowania przewija się w całej książce. Porcje tekstu są zgrupowane zwykle za pomocą znaków **{** na początku grupy i **}** na jej końcu; wówczas efekt działania danej komendy jest ograniczony do tej grupy. Aby komenda działała również poza grupą, należy ją poprzedzić komendą **\global**.

T_EX posiada jeszcze inne mechanizmy grupowania. Istnieją komendy podstawowe: **\beginngroup** i **\endgroup** oraz **\bgroup** i **\egroup**, które mają podobne działanie. Komendy te nie zawsze działają wymiennie: **\beginngroup** i **\endgroup** nie mogą być wykorzystane do ograniczenia argumentów komend (do oznaczenia porcji tekstu, na który działa komenda). Na przykład kombinacja komend **\centerline\beginngroup tekst \endgroup** nie jest poprawna (szczegóły można znaleźć w opisie komendy **\bgroup**).

\halign * 42, 44–45, 48–50, 66–67

Wyrównuje materiał w tabelach, itp. Jest komendą podstawową dla innych wyrównań, jak np. **\settabs** (patrz przykład VI).

\hangafter * 58–59

Wykorzystywana w połączeniu z komendą **\hangindent** w celu wyspecyfikowania liczby wierszy, po której ma nastąpić wcięcie zawieszane (jeśli **\hangafter** jest liczbą dodatnią), lub liczby wierszy, które mają zostać wcięte (jeśli **\hangafter** jest liczbą ujemną). Patrz przykład VIII.

`\hangindent` * 58–59

Określa wielkość zawieszzonego wcięcia, jak to pokazano w przykładzie VIII.

`\hbadness` * 80

Określa karę w poziomym trybie pracy, za odstęp, jakie T_EX może tolerować, zanim wyświetli raport o niepełnym pudełku poziomym. Przetwarzając tekst, T_EX czyta całe akapity i próbuje złożyć je bez dużej liczby nadmiarowych odstępów. Ilość akceptowalnego odstępu określa parametr zwany `\tolerance`. Po złożeniu tekstu T_EX informuje o każdym pudełku `\hbox`, dla którego kara (*badness*) przekracza dopuszczalną wartość `\hbadness`. Parametr domyślny w tym przypadku wynosi 1000 w skali od 0 do 10000. Należy podkreślić, że parametr ten nie wpływa na rozmieszczenie tekstu, określa jedynie, na które pudełka należy zwrócić uwagę.

`\hbox` * 2, 38–41, 54–57, 60–63, 74–83

Komenda ta pozwala na umieszczenie materiału w poziomym pudełku (patrz pudełka). Rozmiar poziomy pudełka `\hbox` (jego szerokość) można określić za pomocą słowa kluczowego `to`. Na przykład wiersz

```
„Dwa                cale”
```

powstał z pliku wejściowego:

```
... Na przykład wiersz \smallskip
  \hbox to 2in{„Dwa\hfil cale”} powstał...
```

Jeśli nie jest określony rozmiar pudełka, przybiera ono szerokość zawartości. Szerokość ta może się zmieniać po wyspecyfikowaniu elastyczności (tj. po określeniu, jak bardzo szerokość pudełka `\hbox` może różnić się od naturalnej). Na przykład wiersz

```
„Ładne dopasowanie”    „Ładne  dopasowanie”
```

powstał z pliku:

```
... Na przykład wiersz\smallskip\line
  {\hbox{„Ładne dopasowanie”}}
  \quad\hbox spread 10pt
  {„Ładne dopasowanie”}\hfil} powstał...
```

`\hbox` normalnie umieszcza materiał w jednym wierszu. Aby materiał znalazł się w kilku wierszach, należy „włożyć” go w `\vbox` i wtedy w `\hbox`. Na przykład

```
Dwa
```

```
Wiersze
```

powstały z pliku:

```
\smallskip
  \hbox{\vbox{Dwa\hfil\break Wiersze}}
  powstały...
```

`\headline` 2, 4, 20–21, 24–25, 34–37

Składa nagłówek (patrz uwagi do str. 4 oraz `\footline`).

`\hfil` * viii–xi, 12–13, 18–21, 44–45, 48–49, 54–55, 60–61, 74–75

Wypełnia wiersz justunkiem (patrz słowa kluczowe, `\fil`).

`\hfill` * 12–13, 44, 50–51, 60–63, 74–75, 82–83

Działa podobnie jak `\hfil`, jest jednak komendą silniejszą. Najczęściej jest wykorzystywana w celu zastąpienia innych komend dotyczących odstępów (patrz `\fil` oraz `1` pod hasłem słowa kluczowe). Na przykład

\hfilneg

Te dwa wiersze

powstały z wpisu:

```
\line{\hfil Te dwa wiersze \hfil}
\line{\hfil powstały z wpisu:\hfill}
```

\hfilneg *

Likwiduje efekt działania komendy `\hfil`.

Na przykład plik wejściowy

tych dwóch wierszy ma postać:

```
\line{\hfil Na przykład plik wejściowy \hfil}
\line{\hfil tych dwóch wierszy ma postać:\hfil\hfilneg}
```

\hfuzz *

Określa, jak dużą część materiału program może umieścić w pudełku poziomym, zanim poda komunikat o przepełnionym pudełku (*overflow box*). W Plain T_EX-u wartość domyślna `\hfuzz=0,1pt`.

\hglue

Poziomy klej (*glue*), który może być wykorzystany, podobnie jak `\hskip`, do umieszczenia odstępu.

Komenda ta działa nawet na początku wiersza; np. plik wejściowy trzech ostatnich wierszy ma postać:

```
do umieszczenia odstępu.\hfil\break
{\hglue 1cm}Komenda ta działa nawet na początku wiersza; np. plik
wejściowy trzech ostatnich wierszy ma postać:
```

Klej umieszczony na początku wiersza za pomocą komendy `\hglue` nie został usunięty. Inaczej jest w przypadku zastosowania komendy `\hskip` (patrz `\hskip`, `\relax`).

\hoffset * [viii–ix](#), [xiv–1](#)

Instrukcja określająca położenie strony w poziomie względem kartki papieru. Jeśli parametr ten ma wartość 0, lewy margines wynosi 1 cal. Dodatnia wartość `\hoffset` przesunęła lewy margines tekstu w prawo, ujemna wartość — w lewo. Początek pliku wejściowego wstępu zawiera komendy dotyczące położenia tekstu na kartce, a komendy formatujące dodatek, przedstawione w uwagach końcowych, demonstrują, w jaki sposób można kontrolować ten parametr.

\hphantom [62–63](#)

`\hphantom{argument}` umieszcza argument w pudełku `\hbox`, mierzy jego szerokość i składa puste pudełko o zmierzonej szerokości, ale zerowej wysokości i głębokości (odsadce). Na przykład, tu `\hphantom{...}` znajduje się odstęp o szerokości odpowiadającej szerokości słowa „wypełnienie”. Komenda ta jest bardzo przydatna (patrz także `\phantom` i `\vphantom`).

\hrule * [vi–vii](#), [12–13](#), [34–37](#), [48–49](#), [58–59](#), [66–69](#), [76–77](#), [80–85](#)

Składa kreskę poziomą (*ruled line*). Patrz kreska (`rule`).

`\hsize` * [viii-ix](#), [xiv-1](#), [66-69](#), [74-77](#), [80-87](#)

Określa poziomy rozmiar (szerokość) strony i pudełek na stronie (patrz str. 1 i przykład XV). Domyślna wartość `\hsize` w Plain T_EX-u to 6,5 cala. Dla pudełek na stronie można wybrać inne wartości `\hsize` i te wartości nie mają wpływu na szerokość strony. W przypadku określania `\hsize` w powiększonym dokumencie, bezpieczniej jest użyć słowa kluczowego `true` (patrz jednostki).

`\hskip` * [11](#), [78-79](#), [80](#), [82-83](#)

Umieszcza klej składający poziomy odstęp określonego rozmiaru. Na przykład T_EX przeskoczy teraz dokładnie o jeden cal. Komenda ta nie zadziała na początku wiersza; a oto plik wejściowy powyższego fragmentu:

```
... przeskoczy teraz dokładnie o jeden{\hskip 1in}cal.\hfil\break
{\hskip 1in}Komenda ta nie zadziała na początku wiersza; a oto...
```

Innymi słowy, T_EX usuwa klej umieszczony przez komendę `\hskip` w miejscach, gdzie wydaje się on niepotrzebny. Na przykład w wyliczeniach typu a) pierwsza jednostka, b) druga jednostka, c) trzecia jednostka, d) czwarta jednostka, e) piąta jednostka, klej nie jest potrzebny na początku czy na końcu wiersza. Plik wejściowy tego akapitu ma postać:

```
... typu{\hskip,185in a)~pierwsza jednostka, \hskip,185in b)~druga
jednostka, \hskip,185in c)~trzecia jednostka, \hskip,185in d)~czwarta
jednostka, \hskip,185in e)~piąta jednostka, ...
```

(patrz `\hglue`, `\null` i `\relax`).

`\hss` *

Ma związek z poziomą elastycznością. Wypełnia wiersz klejem, który może dowolnie kurczyć się (tak jak `\hfil`) lub rozciągać (patrz słowa kluczowe, `\fil`).

`\ht` * [80](#)

Przy składzie strony, znajdujący się na niej materiał T_EX umieszcza w pudełku `\box255` (patrz `\dp`); po wypuszczeniu strony pudełko to jest puste. Wysokość takiego pudełka można uzyskać np. za pomocą wpisu `\ht255`. Wartość ta może zostać wydrukowana komendą `\the\ht255`. W tym miejscu wysokość pudełka `\box255` wynosi 0.0pt, co oznacza, że jest ono puste.

`\ifcase` *

Jedna z wielu komend warunkowych. Pozwala na powiązanie sytuacji z różnymi możliwościami czy przypadkami (patrz `\month`).

`\ifdim` *

Pozwala porównywać wielkości z wymiarami. A oto struktura komendy:

```
\ifdim warunek, instrukcja, kiedy warunek jest prawdziwy
  \else instrukcja, kiedy warunek jest fałszywy \fi
```

W tej książce przedstawiono cztery klasy komend warunkowych, ale w T_EX-u dostępne są jeszcze inne; pełna dyskusja na ten temat znajduje się w rozdziale 20 T_EXbook-a.

`\ifnum` * [68-73](#), [78-81](#)

Podobna do `\ifdim`, ale wykorzystywana do porównywania liczb całkowitych.

\ifodd

\ifodd * 80–81

Sprawdza, czy dana zmienna reprezentuje liczbę nieparzystą (patrz komendy dotyczące numeracji stron w przykładzie XVII lub specyfikacja komend `\headline` oraz `\output` w uwagach końcowych).

\ignorespaces *

Działa w ten sposób, że ignorowane są kolejno po sobie następujące puste spacje. Jest to przydatne przy definiowaniu własnych komend; daje pewność, że w pliku wyjściowym nie pojawią się dodatkowe odstępy.

ignorowanie tekstu

Patrz komentarze.

in vi–vii, xiv–1

Patrz słowa kluczowe.

indeksy 24–29, 36–44, 52–57

Temat ten poruszony jest w przykładach II i V oraz w opisie matematyka: odstępy. Indeksy niekoniecznie muszą być cyframi czy literami i niekoniecznie muszą znajdować się po prawej stronie wyrażenia. Na przykład

$${}_2A^7 \quad \dots \spadesuit \heartsuit \clubsuit \dots \quad k_{[\mu} C_{\nu]} \alpha_{\beta} [\rho k_{\sigma]} k^{\alpha} k^{\beta}.$$

Plik wejściowy ostatniego wiersza ma postać:

```

$$_2A^7
\qqquad\qqquad %ODSTĘP POMIĘDZY WZORAMI
^{\ldots\spadesuit\flat}_{\ldots\diamondsuit\top}
\Re^{\natural\heartsuit\ldots}_{\sharp\clubsuit\ldots}
\qqquad\qqquad %ODSTĘP POMIĘDZY WZORAMI
k_{[\mu}C_{\nu]}\alpha\beta[\rho]k_{\sigma]}k^{\alpha}k^{\beta}.$$.

```

Zamiast znaków specjalnych `_` i `^`, można użyć komend `\sb` i `\sp`.

\indent * 30–31, 54–57

Wcina pierwszy wiersz akapitu. Może być wykorzystana do wcinania wierszy wewnątrz akapitu, które normalnie nie są wcinane (np. pierwszy wiersz po równaniu eksponowanym), lub do dodania ekstra wcięcia pierwszego wiersza akapitu (podwójne wcięcie składa się za pomocą kombinacji `\indent\indent`).

\input * 60–61, 66

Włącza do dokumentu materiał z innych plików. Typowym zastosowaniem tej komendy jest umieszczenie wszystkich komend formatujących w jednym pliku (nazwanym np. *format*) i wprowadzenie komendy `\input format` na początku składanego dokumentu (patrz ramka z plikiem wejściowym na początku dodatku).

interpunkcja

Umieszczanie znaków interpunkcji jest w większości przypadków proste. Mały problem jest z cudzysłowami. Na przykład `,tak'`, `„tak”`, `„tak”` i `„nie”` uzyskano z pliku:

Na przykład `,tak'`, `„tak”`, `{,},,tak'''` i `~, \thinspace,nie'\thinspace'`...

W ostatnim przypadku potrzebny był ekstra odstęp, ponieważ T_EX zostawia większy odstęp przed znakiem „, i za znakiem „’ niż przed „, i za „’ . (Jeśli na klawiaturze nie ma symboli cudzysłowu otwierającego i zamykającego, można je uzyskać za pomocą komend `\lq` i `\rq`.) Dostępne są również znaki interpunkcyjne popularne w innych językach, np. `ı` i `ı̇` (patrz ligatury i przykład IV).

`\it` vi–ix, xiv–1, 8–9, 20–21, 86–87

Składa tekst *kursywą* (`{\it kursywą}`).

`\item` x–xiii, 6–7, 32–33, 36–37, 52–53, 68–69, 72–73, 84–85, 88–89

Składa wyliczenia. Przykłady takich wyliczeń można znaleźć w plikach

- 1) spisu treści,
- 2) przykładu IV,
- 3) dowolnego wyliczenia w tej książce.

Plik wejściowy ostatnich kilku wierszy ma postać:

```
{\parindent20pt \noindent Tworzy wyliczenia. Przykłady takich
wuliczeń można znaleźć w plikach \item{1} spisu treści,
\item{2} przykładu IV, \item{3} dowolnego wyliczenia
w tej książce.\par}
```

Wyliczenie jest wcinane o aktualną wartość `\parindent` (tzn. aktualnego wcięcia akapitu); wartość tego parametru można zmienić (patrz pierwszy wiersz podanego wyżej przykładu).

`\itemitem` 30–31, 88–89

Wykorzystywana do składu

- wyliczeń zagłębionych; przykłady takich wyliczeń znajdują się w plikach wejściowych
 - * cytowanej literatury,
 - * dowolnego wyliczenia zagłębionego w tej książce;
- wyliczeń wciętych o podwójną wartość `\item`.

Plik wejściowy ostatnich wierszy ma postać:

```
{\parindent20pt \noindent Wykorzystywana do składu
\item{\bullet} wyliczeń zagłębionych; przykłady
takich wyliczeń znajdują się w plikach wejściowych
\itemitem{\star} cytowanej literatury lub
\itemitem{\star} dowolnego wyliczenia zagłębionego w tej książce;
\item{\bullet} wyliczeń wciętych o podwójną wartość "\item".\par}
```

jednostki 9

Komendy dotyczące odstępów, takie jak `\hskip`, muszą być podawane w jednostkach rozpoznawanych przez program. W tej książce wykorzystane są cale, centymetry i punkty, ale są jeszcze inne jednostki rozumiane przez T_EX-a. Oto ich lista:

`\joinrel`

cal (in)	
centymetr (cm)	2,54 cm = 1 in
punkt (pt)	72,27 pt = 1 in
pika (pc)	1 pc = 12 pt
duży punkt (bp)	72 bp = 1 in
milimetr (mm)	10 mm = 1 cm
punkt typograficzny (dd)	1157 dd = 1238 pt
cycero (cc)	1 cc = 12 dd
punkt skalowany (sp)	65536 sp = 1 pt

\TeX dokonuje konwersji wszystkich jednostek na `sp` i za ich pomocą liczy rozmiary. Ponieważ $1 \text{ sp} \approx 5 \times 10^{-7} \text{ cm}$, możliwe jest dokładne określenie pozycji danego materiału. \TeX nie przyjmie rozmiarów $\geq 2^{30} \text{ sp}$ ($\approx 575 \text{ cm}$) (patrz także `em` i `ex`).

Jeśli dokument został w całości powiększony, należy uważać przy specyfikowaniu rozmiarów. Aby faktycznie uzyskać np. `3cm`, jako rozmiar określonej wielkości, należy napisać `3 true cm`, co pozwoli uniknąć efektu powiększenia.

`$$\joinrel$`

Patrz matematyka: efekty specjalne.

`\jot`

Parametr najczęściej wykorzystywany do zmiany odstępów w wieloliniowych wyrażeniach eksponowanych. Jego wartość w Plain \TeX -u wynosi `3pt`. Można ją zmienić np. na 300 punktów za pomocą wpisu `\jot=300pt` (patrz `\openup`).

justowanie 86–87

W \TeX -u marginesy tekstu są sztywno określone. W opisach dotyczących komend `\raggedright` i `\raggedleft` przedstawiono sposób wyłączenia takiego wyrównania, a w uwagach końcowych znajduje się nowa definicja komendy `\raggedleft`.

kategorie znaków

Kiedy \TeX przetwarza stronę, w różny sposób traktuje różne znaki: niektóre są drukowane, inne traktowane jako komendy. Na przykład znak `$` zwykle nie jest traktowany jako instrukcja do drukowania, ale jako początek lub koniec trybu matematycznego. Reakcja programu zależy od kodu kategorii danego znaku. Jest 16 kategorii znaków, oznaczanych (kodowanych) liczbami od 0 do 15. Kategorie i standardowe przypisanie znaku do każdej kategorii są następujące:

0 znak sterujący	<code>\</code>	8 frakcja dolna	<code>_</code>
1 początek grupy	<code>{</code>	9 znak ignorowania	<code>null</code>
2 koniec grupy	<code>}</code>	10 odstęp	<code>space</code>
3 tryb matematyczny	<code>\$</code>	11 litera	<code>a,...,Z</code>
4 tabulacja	<code>&</code>	12 inny znak	żaden z pozostałych
5 koniec wiersza	<code>return</code>	13 znak aktywny	<code>~</code>
6 parametr	<code>#</code>	14 komentarz	<code>%</code>
7 frakcja górna	<code>^</code>	15 znak błędny	<code>delete</code>

Kategorie znaków nie powinny być mylone z kodami znaków; kody znaków identyfikują znak, kategoria znaku określa rolę, jaką ten znak pełni w danym momencie. Każdy znak w \TeX -u jest widziany wewnętrznie jako para liczb: kod znaku i kategoria znaku. Oznaczenie kategorii dokonuje się podczas przetwarzania pliku w \TeX -u; następuje wówczas przełączenie kategorii potrzebnych do nowych, zaawansowanych aplikacji.

Przełączanie kategorii (przedstawione w uwagach końcowych, w definicji komend `\literal` i `\ignore`) powoduje, że znaki grają inną niż standardowo rolę. Na przykład dowolny znak oznaczony kategorią 11 będzie zachowywał się jak litera i może zostać wykorzystany w nazwie komendy. Technika ta jest stosowana jako środek ostrożności przy tworzeniu pakietów nowych komend: znak @, który normalnie posiada kategorię 12, zostaje tymczasowo przypisany do kategorii 11 i wykorzystany w nazwach komend stosowanych wewnętrznie w pakiecie.

Kategorie znaków są przypisywane do znaków za pomocą komendy `\catcode`. Na przykład `\catcode37=1` umieszcza znak odpowiadający znakowi dziesiętnemu o kodzie 37 w kategorii 1, tzn. znak ten będzie teraz grał rolę symbolu grupującego {. Jeśli nie znamy kodów znaków lub jeśli męczące jest ich szukanie, to możemy wykorzystać technikę omówioną przy końcu opisu pojęcia kody znaków, służącą do automatycznego wywoływania kodów. Według tej techniki komenda `\$` wywoła kod znakowy dla \$ itd. Komenda `\catcode'\$=4` spowoduje, że znak \$ będzie grał rolę tabulatora & (znak & będzie grał taką samą rolę, dopóki jego znaczenie nie zostanie zmienione).

`\kern` * 74–79

Wstawia poziomy albo pionowy odstęp w zależności od kontekstu i trybu pracy \TeX -a (patrz tryby pracy). \TeX nie dzieli wiersza w miejscu odstępu poziomego stworzonego przez `\kern` (w przeciwieństwie do odstępu uzyskanego np. komendą `\hskip`).

kerning

Przemieszczanie znaków, dosuwanie ich do siebie, w celu bardziej estetycznego dopasowania (pozbycia się zbyt dużego odstępu między nimi). Porównaj np. parę liter z kerningiem **VA** i bez kerningu **VA**. \TeX generalnie nie ma problemów z kerningiem (światłem międzyliterowym), można jednak dokonywać tu dowolnych zmian. Na przykład plik wejściowy dla **VA** i **RA** ma postać:

```
{\bf V\kern-3pt A} i R\kern-2,1pt A
```

klej (glue)

Substancja bezbarwna, bezwonna, bezpostaciowa, przenikająca każdą składaną stronę. Konstrukcja ta jest wykorzystana do kontroli justunku na stronie i określenia jego zdolności do kurczenia się czy rozciągania. Można odłożyć klej, korzystając bezpośrednio z komend `\vglue` i `\hglue` lub `\vskip` czy `\hskip`. Klej „położony” za pomocą komend `skip` jest w szczególnych sytuacjach usuwany przez program (`\hskip` jest ignorowana na początku wiersza, wewnątrz akapitu, `\vskip` — u góry strony), natomiast klej umieszczony za pomocą komend `\vglue` czy `\hglue` nie znika (patrz osobne opisy dotyczące tych komend).

Zdolność kurczenia czy rozciągania kleju jest określana za pomocą słów kluczowych `plus` i `minus` (patrz także `\bigskip`, `\medskip`, `\smallskip` i `\parskip`).

kody znaków

\TeX wewnętrznie określa znaki przy użyciu kodów liczbowych. Kod może być wyrażony w różny sposób: w notacji dziesiętnej, ósemkowej, czy szesnastkowej. Można wywołać znak przez odniesienie się do jego kodu za pomocą komendy `\char` (tabele kodów są umieszczone w dodatku F książki *TEXbook*, a poniżej znajduje się ich wersja dziesiętna). Liczby w kodzie ósemkowym są poprzedzone znakiem ', a w kodzie szesnastkowym — znakiem ". Kod znaku w dowolnej z tabel uzyskamy przez

poszukanie odnośnika w wierszu i w kolumnie. Na przykład etykieta wiersza '01x i etykieta kolumny '5 odpowiada kodowi znakowemu '015, w skrócie '15. Jeśli określa się znaki za pomocą ich kodów, należy także wyspecyfikować krój pisma, ponieważ różne znaki mogą zajmować tę samą pozycję w tabeli i oznaczać różne kroje pisma. I tak `{\sl\char'15}` oznacza *fl*, podczas gdy `{\tt\char'15}` — znak '. W innym kroju jest to równoważne wpisowi `\char'13`, ponieważ w kodzie ósemkowym liczba 15 odpowiada liczbie 13 w kodzie dziesiętnym.

To udogodnienie jest bardzo pożyteczne, ponieważ pozwala drukować znaki, które nie występują na klawiaturze, lub te, które występują, ale są traktowane jak komendy. Na przykład znak " jest wykorzystywany w dodatku jako komenda, a w tym miejscu został wydrukowany za pomocą komendy `{\tt\char'42}`.

Znaki i tabelę ich kodów można szybko wygenerować za pomocą komendy zdefiniowanej poniżej. Komenda ta drukuje wszystkie znaki danego kroju pisma, wraz z ich kodami znakowymi. Na przykład `\chartable{plr10}` działa następująco:

Znaki fontu *plr10* z kodami dziesiętnymi:

0: Γ	1: Δ	2: Θ	3: Λ	4: Ξ	5: Π	6: Σ
7: Υ	8: Φ	9: Ψ	10: Ω	11: ff	12: fi	13: fl
14: ffi	15: ffl	16: ı	17: j	18: `	19: ´	20: ˘
21: ˘	22: ¯	23: °	24: ˙	25: ß	26: æ	27: œ
28: ø	29: Æ	30: Œ	31: Ø	32: -	33: !	34: ”
35: #	36: \$	37: %	38: &	39: ’	40: (41:)
42: *	43: +	44: ,	45: -	46: .	47: /	48: 0
49: 1	50: 2	51: 3	52: 4	53: 5	54: 6	55: 7
56: 8	57: 9	58: :	59: ;	60: j	61: =	62: ĺ
63: ?	64: @	65: A	66: B	67: C	68: D	69: E
70: F	71: G	72: H	73: I	74: J	75: K	76: L
77: M	78: N	79: O	80: P	81: Q	82: R	83: S
84: T	85: U	86: V	87: W	88: X	89: Y	90: Z
91: [92: “	93:]	94: ^	95: ˙	96: ‘	97: a
98: b	99: c	100: d	101: e	102: f	103: g	104: h
105: i	106: j	107: k	108: l	109: m	110: n	111: o
112: p	113: q	114: r	115: s	116: t	117: u	118: v
119: w	120: x	121: y	122: z	123: –	124: —	125: ”
126: ~	127: ˆ					

```
\def\chartable #1{\smallbreak\vbox{\noindent
  {\sl Znaki fontu {\it #1} z kodami dziesiętnymi\}/:
  \vskip0pt \raggedright \hbadness5000 \tolerance10000 \font\ft=#1 \ft
  \baselineskip=14pt \ifdim\baselineskip<3,25ex \baselineskip=3,25ex\fi
  \count255=0 \dimen0=1-pt
  \loop \setbox0=\hbox{\char\count255} %Wyszukanie najszerszego znaku.
    \ifdim\wd0>\dimen0 \dimen0=\wd0 \fi \advance\count255 by1
    \ifnum\count255<128 \repeat
  \count255=0 \advance\dimen0 by 25pt \noindent
  \loop\hbox to\dimen0{\hbox to23pt{\hfil
    \rm\the\count255:\ } \char\count255\hfil}
    \advance\count255 by1 \ifnum\count255<128 \quad \repeat
  \smallbreak} \font\tenrm=plr10 }
```

Aby nie przeszukiwać tabeli kodów, można w celu znalezienia kodu danego znaku wykorzystać symbol ‘, który „pyta” o kod znakowy. Na przykład kod dziesiętny znaku d wynosi 100; komenda `\char100` jest więc równoważna komendzie `\char‘d`. Można także korzystać z komend pojedynczych znaków, np. `\%`: kod dziesiętny `%` wynosi 37; `\char37` jest równoważne `\char‘\%`. To ostatnie zastosowanie jest przydatne przy wywoływaniu kodów znakowych dla komend znakowych.

komendy (polecenia, instrukcje)

Nazwy komend w \TeX -u są tak dobrane, aby sugerować efekt ich działania. Ułatwia to ich zapamiętywanie oraz umiejscowienie przy czytaniu pliku wejściowego. Znaczenie niektórych komend nie jest oczywiste — głównie tych kontrolujących odstęp czy komend znakowych (patrz lista komend znakowych). Pierwsza strona dodatku zawiera informacje dotyczące klas komend. Wykaz komend zdefiniowanych w tej książce umieszczony jest pod hasłem nowe komendy w podręczniku.

komendy podstawowe

Na podstawowych komendach \TeX -a oparte są wszystkie pozostałe komendy (patrz Plain \TeX). W książce tej omówiono zarówno podstawowe, jak i złożone komendy Plain \TeX -a. Aby łatwiej je było rozpoznawać w dodatku, podstawowe komendy poprzedzone są znakiem * (tam gdzie są wymieniane, a nie omawiane).

komentarze [viii–ix](#), [6–7](#), [20–21](#), [44–45](#), [76–77](#)

Standardowym znakiem komentarza jest znak `%`. \TeX ignoruje wszystko, co znajduje się za tym znakiem, i czyta plik od nowego wiersza. Przy komentarzach wielolinowych, męczące jest wstawianie takiego znaku na początku każdego wiersza. Aby uporać się z takimi przypadkami, zdefiniowano nową komendę `\ignore`:

```
%UWAGA! W tej definicji wykorzystano rejestr pomocniczy \count255.
\def\ignore {\begingroup \count255=0
  \loop \catcode\count255=14 %Wszystkie znaki stają się znakami
                                %komentarza
  \advance\count255 by1 \ifnum\count255<128
  \repeat \catcode‘\!=0 } %Znak ! staje się znakiem sterującym
{\catcode‘\!=0 !gdef!E{!endgroup}}%Koniec definicji komendy
```

[*Wyjaśnienie techniczne (można je pominąć)*: Komenda ta rezerwuje jeden znak do specjalnych celów i zmienia kategorie wszystkich pozostałych znaków na 14 (kategoria komentarza). Po wywołaniu komendy, \TeX traktuje każdy następny wiersz jako komentarz, do momentu, kiedy na początku wiersza pojawi się znak sterujący. W tym przypadku jest to `!`, który z reguły nigdy nie pojawia się sam na początku wiersza, chyba że rozmyślnie zostanie tam umieszczony. Gra wtedy rolę znaku sterującego (`\ backslash-a`); nowa komenda `!E` określa koniec tekstu, który ma być ignorowany. Musi ona być umieszczona na początku wiersza, w innym przypadku \TeX jej nie zobaczy.] A oto przykład wykorzystania tej komendy:

```
\ignore
Od tego momentu \TeX\ będzie ignorował cały tekst dopóki nie napotka
linii rozpoczynającej się komendą !E. Komenda ta uruchomi normalny tryb
pracy \TeX-a. Dopóki \TeX\ nie napotka komendy !E
żadna z komend nie będzie respektowana:
\vskip 5000in, \end, \bye, itd.
Można zatem szaleć z pisownią i składnią, ponieważ \TeX\ „nic nie
widzi”. Np. \centerline{próba}! Bardzo odważnie, prawda?
!E
```

kreska (rule)

`\ignore` jest przydatna także w innym kontekście. Przy pracy nad długim rękopisem czasem wystarczy, aby \TeX przetwarzał tylko część materiału, np. skomplikowaną tabelę. Można wówczas wykorzystać komendę `\ignore` w celu wskazania \TeX -owi, którą część materiału ma pominąć, a którą przetwarzać. Należy podkreślić fakt, że znak **!** może pojawić się w zdaniu jako zwykły znak interpunkcyjny (nawet jeśli ma ono być ignorowane), ponieważ staje się on znakiem sterującym dopiero wtedy, kiedy pojawia się na początku wiersza. Gdyby z jakichkolwiek przyczyn plik zawierał wiersze rozpoczynające się w ten sposób, należałoby przebudować tekst lub zmienić definicję komendy `\ignore`.

kreska (rule)

Termin drukarski. Kreski mogą być poziome (`\hrule`) lub pionowe (`\vrule`). Obydwa rodzaje mają wysokość, głębokość i szerokość. Dla dowolnej `\hrule` wartości domyślne wynoszą: *wysokość 0,4 pt, głębokość (odsadka) 0 pt, szerokość zależy od aktualnych warunków*. Te specyfikacje mogą być zmieniane. Na przykład kreski



powstały dzięki komendom:

```
Na przykład kreski\medskip
\hrule
\bigskip
\hrule width3cm height1pt
\bigskip
\hrule width,5pt height,5cm
\bigskip
```

Kreski takie mogą być składane tylko pomiędzy wierszami tekstu. Można także stosować kreski pionowe. Mogą one być składane w środku wiersza: `|` lub `|` lub `|` lub `_____` lub `■`. Plik wejściowy tego fragmentu ma postać:

```
w środku wiersza:
\vrule\ lub
\vrule height,4cm width,5pt depth0pt\ lub
\vrule height0pt width,7pt depth,3cm\ lub
\vrule height,5pt width1cm depth0pt\ lub
\vrule height10pt width4pt depth-6pt. Plik wejściowy...
```

Domyślne specyfikacje w tym przypadku to: *wysokość i głębokość zależnie od aktualnych warunków, szerokość 0,4pt*.

Można także obramować tekst. Komendy służące do tego celu przedstawiono w przykładzie XVI.

Patrz także `\relax`.

kroje pisma (typefaces) 8–9, 14–15, 64–65, 80–81

Patrz fonty.

kursywa

Patrz `\it`.

\last... *

Klasa komend, których przykładami są `\lastbox`, `\lastskip` i `\lastkern`. Odnoszą się (odpowiednio) do ostatnio zbudowanego pudełka (hbox-a czy vbox-a), do ostatniego *skip*-u czy *kern*-u. Na przykład 3.33333pt plus 1.66666pt minus 1.11111pt jest aktualną wartością parametru `\lastskip`; określa tu odstęp między słowami. Plik wejściowy tego fragmentu ma postać:

```
... Na przykład \the\lastskip\ jest aktualną wartością...
```

Komenda `\lastbox` została wykorzystana w definicji nowej komendy `\table`, omówionej pod hasłem `tabele`.

LaTeX

Powszechnie wykorzystywany pakiet komend oparty na komendach podstawowych TeX-a, ułatwiający pracę przy standardowym składzie tekstu, jak np. przy składzie dwukolumnowym. Komendy LaTeX-a nie zawsze są kompatybilne z komendami Plain TeX-a, tak więc należy ostrożnie przestawiać się z jednego pakietu na drugi.

\leaders * 74–75, 82–83

Pozwala na powtarzanie określonego desenia (wypełnienia). Patrz przykład XV.

\$\$\left\$ * 40–43, 60–63

Składa lewe ograniczniki (np. klamry, nawiasy, itd.) o rozmiarze właściwym dla dużych wzorów (patrz przykład IX). `\left` jest wykorzystywana w równaniach eksponowanych, zawsze w połączeniu z komendą `\right`: TeX oczekuje wtedy pary komend `\left`, `\right`. Symbole zastosowane po tych komendach nie muszą być równoważnymi parami: `\left(` może być połączone z `\right]` czy nawet `\right[`, jeśli taki niezwykły efekt jest wymagany. Gdy potrzebny jest tylko jeden z ograniczników (lewy bądź prawy), należy go połączyć odpowiednio z komendą `\left.` lub `\right..`

\leftline 84–85

Wyrównuje tekst lewostronnie.

Plik wejściowy wiersza powyżej ma postać:

```
\leftline{Wyrównuje tekst lewostronnie.}
```

Nie jest to komenda często wykorzystywana, gdyż tekst zwykle jest właśnie w ten sposób wyrównywany, ale jest przydatna w pewnych sytuacjach (patrz `\line`).

\leftskip * 20–23, 70–71

Przemieszcza lewy margines tekstu (patrz przykład I). Wartość `\leftskip` wykorzystywana w składzie akapitu jest komendą działającą na końcu danego akapitu (patrz akapity).

\$\$\leqalignno\$ 54–55

Wyrównuje równania, umieszczając ich numery z lewej strony (patrz przykład VII).

\$\$\leqno\$ * 52–53

Działa podobnie jak `\eqno`, z tym że numer równania umieszcza z lewej strony.

`\let``\let` * 86–87

Przypisuje daną komendę innej komendzie czy pojedynczemu znakowi. Na przykład definicja komendy `\bgroup` w Plain T_EX-u jest następująca: `\let\bgroup={. \let` działa trochę inaczej niż `\def`: po wpisaniu `\let\ a=\b`, `\a` przyjmie wartość `\b` aktualną w momencie wywołania `\let`, nawet jeśli `\b` zostanie później przeddefiniowane. `\def\ a{\b}` działa w ten sposób, że `\a` przyjmuje najbardziej aktualną wartość `\b`, tj. `\a` zmienia swą wartość, jeśli zmienia ją `\b`. Patrz również `\endgraf`.

ligatury

Lista ligatur znajduje się pod odpowiednim hasłem w dodatku. Można się ich pozbyć pisząc np. fi `{f}i` zamiast fi `fi`.

`\limits` * 52–55, 60–61

Patrz matematyka: operacje.

`\line` viii–ix, 12–13, 20–21, 34–37

Umieszcza tekst w wierszu o długości `\hsize`. Jeśli tekst nie wypełnia wiersza, należy umieścić także komendę `\hfil`, aby uniknąć komunikatu o niepełnym pudełku (*underfull box*). Komendy takie, jak `\centerline`, `\leftline` i `\rightline`, są zdefiniowane w T_EX-u za pomocą komendy `\line`; jeśli którakolwiek z tych komend jest wywołana w środku akapitu, wiersz ją poprzedzający musi być najpierw wyraźnie zakończony (np. za pomocą komend `\hfil\break`).

Przykładowo, plik wejściowy ostatnich wierszy ma postać:

```
"\hfil\break"). \hfil\break
\line{Przykładowo, plik wejściowy ostatnich wierszy ma postać:\hfil}
```

Jeśli komenda `\line` lub dowolna komenda na niej oparta (np. `\centerline`) znajdzie się w pierwszym wierszu nowego akapitu, następny wiersz również zostanie wcięty (chyba że użyta będzie komenda `\noindent`). A zatem T_EX nie traktuje komendy `\line` jako znaku początku akapitu.

`\lineskip` *

Kontroluje odstęp, jaki jest umieszczany między dwoma wierszami tekstu, w przypadku gdy wartość `\baselineskip` wydaje się zbyt mała, pudełka zbyt wysokie lub głębokie (patrz `\lineskiplimit`). Plain T_EX domyślnie ustawia wartość `\lineskip` na `1pt` (patrz także odstępy między wierszami).

`\lineskiplimit` *

Parametr określający, jak blisko siebie mogą znaleźć się dwa sąsiednie wiersze (patrz `\lineskip`). Wartość domyślna tego parametru w Plain T_EX-u wynosi `0pt` (patrz odstępy między wierszami). T_EX, składając wiersze, próbuje wstawić między nimi odległość równą `\baselineskip`. Jeśli w rzeczywistości klej pomiędzy wierszami ma wartość mniejszą niż `\lineskiplimit`, zamiast `\baselineskip` wstawiany jest `\lineskip`.

linie tekstu (wiersze)

Informacje dotyczące kresek, takich jak np. ta: _____, można znaleźć pod hasłem kreska (rule). Informacji na temat linii tekstu (wierszy) należy szukać pod osobnymi wpisami w dodatku (np. łamanie wierszy, odstępy między wierszami itd.).

lista komend jednoznakowych

Komendy: `\'`, `\'`, `\'`, `\.`, `\~`, `\^`, `\b`, `\c`, `\d`, `\H`, `\t`, `\u` i `\v` składają akcenty (patrz opis poniżej). [W oryginale była jeszcze wymieniona w tym miejscu komenda `\=` (w Plain TeX-u składająca akcent, np. ā), której w MeX-u został przypisany dywiz — przyp. tłum.].

Komendy

`\#`, `\$`, `\%`, `\&`, `\{`, `\}`, `\|` oraz `_`

składają, odpowiednio:

`#`, `$`, `%`, `&`, `{`, `}`, `||` oraz `_`.

Efekt działania komend

`\i`, `\j`, `\l`, `\L`, `\o`, `\O`, `\P` oraz `\S`

został omówiony pod hasłem znaki specjalne.

`\` (`\` i spacja), `\/`, `\>`, `\,`, `\;` oraz `!\`

są komendami dotyczącymi odstępów (patrz odstęp oraz matematyka: odstęp).

I na koniec:

`\+` rozpoczyna nowy wiersz w wyrównaniu za pomocą komendy `\settabs`;

`\-` sugeruje miejsca podziału (patrz dzielenie wyrazów);

`*` sugeruje złamanie wiersza na iloczynnie (patrz matematyka: łamanie wierszy).

Uwaga: `\`, `\/` oraz `\-` to komendy podstawowe.

lista komend znakowych 6–7

`\` `{` `}` `$` `&` `^` `_` `%` `~` `#`

Ich znaczenie wyjaśnione zostało we wstępie (patrz także kategorie znaków). Kilka kombinacji tych znaków odgrywa specjalną rolę:

`$$` rozpoczyna i kończy tryb matematyczny eksponowany;

`&&` tworzy powtarzalną preambułę w tabelach (patrz tabela 3 w przykładzie VI);

`##` jest wykorzystana zamiast pojedynczego znaku `#` przy definiowaniu komend zawierających wyrównanie (patrz definicja komendy `\address` w przykładzie XI lub komendy `\table` pod hasłem tabele).

lista ligatur

Ligatura to kombinacja dwóch lub więcej znaków, drukowana jako pojedynczy znak. Następujące kombinacje są traktowane przez TeX-a jako ligatury w większości krajów pisma (znak, jaki składają, jest ujęty w nawiasy):

`‘` (`“`), `’` (`”`), `--` (`—`), `---` (`---`), `?‘` (`?`), `!‘` (`!`),
`ff` (`ff`), `fi` (`fi`), `fl` (`fl`), `ffi` (`ffi`) oraz `ffl` (`ffl`),

\llap 36–37

Pozwala na nałożenie elementu na inny element z lewej strony (nakładanie w lewo). Na przykład `a\llap/` składa znak \aleph (patrz także `\rlap`).

\loop 68–69, 72–73, 78–79

Komenda wykorzystana w połączeniu z `\repeat` do powtarzania pewnych działań. Na przykład `\loop A \if... B\repeat` oznacza, że TeX wykona sekwencję *A* i wtedy sprawdzi warunek `\if...`; jeśli warunek ten jest prawdziwy, TeX wykona sekwencję *B* i powtórzy proces; jeśli zaś warunek jest fałszywy, TeX zakończy powtarzanie.

Warunek `\if...` może być dowolną słuszną komendą warunkową T_EX-a (patrz np. `\ifdim`), ale bez znacznika `\fi` (jest on wbudowany w `\repeat`). Należy być ostrożnym w przypadku pętli zagłębionych (pętla wewnątrz pętli). Trzeba wówczas stosować nawiasy `{ i }` w celu odseparowania obu pętli.

\looseness *

Parametr wymuszający skład akapitu dłuższego lub krótszego (w jednostkach ilości wierszy) niż normalnie złożony. Na przykład `\looseness=2` powoduje, że T_EX próbuje złożyć akapit o dwa wiersze dłuższy, niż sam złożyłby, a `\looseness=-1` — o jeden wiersz krótszy. Próbuje, ponieważ T_EX nie może przekroczyć ilości pustego odstępu dozwolonego przez parametr `\tolerance`. Czemu służy ta komenda? Jest kilka jej zastosowań. Na przykład do likwidacji pojedynczych wierszy u dołu lub u góry strony przy składzie książek czy czasopism. Komenda ta może być wykorzystana także w połączeniu z `\prevgraf` objaśnioną poniżej. Przed tym akapitem umieszczono komendę `\looseness=1`, bez której miałby on 9 wierszy. (I to prawda!)

Plik wejściowy pierwszego i ostatniego wiersza akapitu ma postać:

```
\looseness=1 Parametr wymuszający skład akapitu
...
on 9 wierszy. (I~to~prawda!)
```

Wiązanie pomiędzy ostatnimi dwoma słowami (`~`) w przypadku zwiększania liczby wierszy w akapicie (komendą `\looseness` z dodatnim parametrem) blokuje T_EX-owi najłatwiejszą drogę zwiększenia liczby wierszy poprzez przesunięcie ostatniego wyrazu do następnego wiersza.

\lower * 60–61, 74–75

Obniża pudełka. Szczegółowe informacje można znaleźć pod hasłem `\raise`.

\lowercase *

Składa tekst małymi literami; np. plik wejściowy tego fragmentu ma postać:

```
S\lowercase{KŁADA TEKST MAŁYMI LITERAMI; NP. PLIK WEJŚCIOWY TEGO
FRAGMENTU MA POSTAĆ:}
```

łamanie stron (kolumn tekstu) viii–ix, 4–5, 8–9, 18–19

Złamanie strony można zabronić komendą `\nobreak`, gdy T_EX znajduje się w pionowym trybie pracy (patrz tryby pracy); wymusić je zaś można komendą `\break` (w pionowym trybie pracy) lub `\eject` (w dowolnym z trybów pracy programu, ale nie wewnątrz pudełka). Przy wywoływaniu łamania strony za pomocą komend `\break` lub `\eject` należy pamiętać o umieszczeniu wcześniej komendy `\vfil`, w przeciwnym razie pojawi się komunikat `underfull vbox`. Wymuszone złamanie strony powoduje także podział akapitu. Aby dokonać podziału strony na końcu bieżącego wiersza, z kontynuacją akapitu na następnej stronie, należy zastosować komendę `\vadjust{\vfil\eject}`.

Można sugerować T_EX-owi możliwe punkty łamania strony za pomocą komend `\smallbreak`, `\medbreak`, `\bigbreak`, `\filbreak` czy `\goodbreak`. W pierwszych trzech przypadkach, jeśli T_EX nie złamie strony w danym miejscu, wstawi odstęp pionowy o odpowiednim rozmiarze (`\smallskip`, `\medskip` lub `\bigskip`). Na pewno T_EX złamie stronę w miejscu wskazanym komendą `\filbreak` (komenda ta działa jak `\vfil\break`), chyba że znajdzie jeszcze jedną komendę `\filbreak` przed

punktem, w którym ma normalnie złamać stronę. Komenda `\goodbreak` sugeruje dobre miejsce do złamania strony, bez szczególnych restrykcji w razie nieuwzględnienia jej. `\bigbreak` jest wykorzystana w kilku miejscach tej książki, a komendy formatujące dodatek (pokazane w uwagach końcowych) zawierają przeddefiniowanie komendy `\par`, która wykorzystuje w Plain TeX-u komendę `\goodbreak`.

macierze 40–41

Komenda `\pmatrix` składa macierz w nawiasach węższych (klamrach), `\matrix` pozwala na wyspecyfikowanie swoich własnych ograniczników (patrz przykład V). W TeX-u dostępna jest także komenda `\bordermatrix`, pozwalająca na skład następujących wyrażeń:

$$\begin{array}{c} \triangle \quad \heartsuit \quad \wp \\ \mathfrak{S} \left(\begin{array}{ccc} a & b & c \\ d & e & f \\ 1 & \mu & -73 \end{array} \right) \\ \aleph \end{array}$$

uzyskanych z pliku:

```
$$\bordermatrix{\triangle&\heartsuit&\wp\cr
\Im&a&b&c\cr \ell&d&e&f\cr \aleph&1&\mu&-73\cr}$$
```

Za pomocą komendy `\matrix` można złożyć macierz z niewidzialnym (lewym bądź prawym) ogranicznikiem, poprzez wpisanie, odpowiednio, `\left.` lub `\right..`

`\magnification` vii–ix, xiv–1

Powiększa cały dokument. Urządzenia drukujące często przyjmują tylko wybrane stopnie pisma, które można uzyskać za pomocą komendy `\magstep` (patrz plik wejściowy str. 1). Takie komendy muszą pojawiać się zaraz na początku pliku, ponieważ inaczej TeX zacznie składać materiał bez powiększenia. W opisie hasła jednostki znajduje się dyskusja na temat określania rozmiaru powiększanego dokumentu.

`\magstep` viii–ix, xiv–1, 64–65, 80–81

Określa kolejne kroki powiększenia. Plain TeX ma wbudowane następujące rozmiary powiększeń (w skali, w której rozmiar bez powiększenia, `\magstep0`, wynosi 1000):

<code>magstephalf:</code>	1095	<code>magstep1:</code>	1200
<code>magstep2:</code>	1440	<code>magstep3:</code>	1728
<code>magstep4:</code>	2074	<code>magstep5:</code>	2488

Kolejne rozmiary uzyskuje się przez pomnożenie liczby 1000 przez odpowiednią potęgę liczby 1,2. `\magstep1` określa 20-procentowe powiększenie. Rozmiary te wykorzystywane są do powiększania całych dokumentów, np. za pomocą komendy `\magnification`, lub do powiększania fontów (patrz powiększanie).

makra

Komputerowy żargon określający nowe komendy, zdefiniowane przez użytkownika za pomocą komend już istniejących (patrz definicje).

marginesy

TeX składa strony z 1-calowymi marginesami: górnym i lewym. Domyślny rozmiar strony (tj. rozmiar materiału na stronie) to $6,5 \times 8,9$ cali, co daje również 1 cal marginesu dolnego i prawego (przy rozmiarze kartki $8,5 \times 11$ cali; margines dolny obejmuje stopkę).

\mark

Marginesy można regulować komendami `\hsize`, `\vsize`, `\hoffset` i `\voffset`. Pierwsze dwie zmieniają rozmiar materiału drukowanego na stronie, następne kontrolują jego położenie. Za pomocą tych komend można dowolnie określać marginesy. Na przykład 1,5-calowy lewy margines i 3-calowy margines prawy na stronie $8,5 \times 11$ cali uzyskuje się ustawiając `\hsize 4in` i `\hoffset ,5in`. Można także zmieniać marginesy w dokumencie tymczasowo, za pomocą komend `\leftskip` i `\rightskip`.

\mark *

Umieszczana w dokumencie w celu zaznaczenia tekstu, do którego ma być odwołanie na kolejnych stronach (np. nagłówek każdego wpisu w dodatku jest „zaznaczony” za pomocą tej komendy; patrz komendy formatujące dodatek). Podczas składania stron, komendy `\topmark`, `\firstmark` i `\botmark` wybierają taki zaznaczony tekst ze strony. `\topmark` — ostatnio zaznaczony tekst bezpośrednio przed daną stroną, `\firstmark` — pierwszy zaznaczony na stronie tekst, a `\botmark` — ostatni zaznaczony na stronie tekst (więc `\botmark` jednej strony może być tekstem `\topmark` strony następnej). Małe nagłówki pod numerami stron dodatku są tekstem `\firstmark` (w przypadku stron „lewych”) lub `\botmark` (w przypadku „prawych”).

matematyka 24–33, 38–43, 52–57, 60–63

Wewnątrz opisów powiązanych z hasłem **matematyka** wymienione zostały wszystkie symbole specjalne, powszechnie dostępne w trybie matematycznym. Przedstawiono także kilka efektów specjalnych oraz omówiono dopasowanie odstępów w trybie matematycznym. Inne aspekty składu wyrażeń matematycznych są wymienione pod osobnymi nagłówkami (np. ułamki, wyrównywanie równań, macierze, pierwiastki). Wszystkie symbole matematyczne należy ograniczać znakami `$` (tylko kilka z nich jest dostępnych poza trybem matematycznym; umieszczane są w tabelach bez znaków `$`). Równania, które chcemy eksponować w osobnym wierszu, muszą znajdować się pomiędzy podwójnymi znakami `$`. Styl eksponowany automatycznie wyśrodkowuje równanie, składa je w większym formacie i umieszcza dodatkowy odstęp nad i pod równaniem.

matematyka: akcenty 52–57

\hat{o}	<code>\hat o\$</code>	\check{o}	<code>\check o\$</code>	\tilde{o}	<code>\tilde o\$</code>	\acute{o}	<code>\acute o\$</code>
\grave{o}	<code>\grave o\$</code>	\dot{o}	<code>\dot o\$</code>	\ddot{o}	<code>\ddot o\$</code>	\breve{o}	<code>\breve o\$</code>
\bar{o}	<code>\bar o\$</code>	\vec{o}	<code>\vec o\$</code>				

Podobnie jak w przypadku akcentów w tekście, aby umieścić znak akcentu nad literami *i* lub *j*, należy je najpierw pozbawić kropek. Komendy `\imath` i `\jmath` składają, odpowiednio, *i* i *j*. W trybie matematycznym dostępne są także „szerokie” akcenty, zmieniające rozmiar w zależności od znajdującego się pod nimi wyrażenia:

\overline{o}	<code>\overline o\$</code>	\overline{abcde}	<code>\overline{abcde}\$</code>
\underline{o}	<code>\underline o\$</code>	\underline{abcde}	<code>\underline{abcde}\$</code>
\widehat{ab}	<code>\widehat{ab}\$</code>	\widehat{abcde}	<code>\widehat{abcde}\$</code>
\widetilde{ab}	<code>\widetilde{ab}\$</code>	\widetilde{abcde}	<code>\widetilde{abcde}\$</code>

Podczas gdy `\overline` i `\underline` tworzą kreski, które „rosną” wraz z rozmiarem dużych wyrażeń, kolejne dwa akcenty mogą objąć maksymalnie trzy znaki. Można akcentować wyrażenia już akcentowane, ale należy zwrócić uwagę na umieszczenie akcentu. Dyskusja na ten temat znajduje się poniżej, pod hasłem **matematyka**: odstępy (patrz także `\over` i `\under`).

matematyka: alfabet grecki 38–43, 54–57

α <code>\alpha</code>	β <code>\beta</code>	γ <code>\gamma</code>	δ <code>\delta</code>
ϵ <code>\epsilon</code>	ε <code>\varepsilon</code>	ζ <code>\zeta</code>	η <code>\eta</code>
θ <code>\theta</code>	ϑ <code>\vartheta</code>	ι <code>\iota</code>	κ <code>\kappa</code>
λ <code>\lambda</code>	μ <code>\mu</code>	ν <code>\nu</code>	ξ <code>\xi</code>
o <code>o</code>	π <code>\pi</code>	φ <code>\varphi</code>	ρ <code>\rho</code>
ϱ <code>\varrho</code>	σ <code>\sigma</code>	ς <code>\varsigma</code>	τ <code>\tau</code>
υ <code>\upsilon</code>	ϕ <code>\phi</code>	φ <code>\varphi</code>	χ <code>\chi</code>
ψ <code>\psi</code>	ω <code>\omega</code>		
Γ <code>\Gamma</code>	Δ <code>\Delta</code>	Λ <code>\Lambda</code>	Θ <code>\Theta</code>
Ξ <code>\Xi</code>	Π <code>\Pi</code>	Υ <code>\Upsilon</code>	Σ <code>\Sigma</code>
Φ <code>\Phi</code>	Ψ <code>\Psi</code>	Ω <code>\Omega</code>	

Pochyle wersaliki można uzyskać przez poprzedzenie komendy przedrostkiem `\mit`. Na przykład `\mit\Gamma` składa Γ .

matematyka: efekty specjalne

Użytkownik TeX-a ma możliwość umieszczania symboli nad innymi symbolami:

```

 $\overset{\mu,\nu}{\text{def}}$  \{\buildrel\mu,\nu\over\longmapsto\}
 $\overset{\text{def}}{\propto}$  \{\buildrel\rm def\over\propto\}
 $\overset{\circ}{g}_{\mu\nu}$  \{\buildrel\circ\over g\}_{\mu\nu}

```

Inna z komend `\joinrel` pozwala łączyć symbole relacji. Oto kilka przykładów: symbol \longleftrightarrow powstaje jako skutek działania komendy `\lonleft\rightarrow`, zdefiniowanej:

```
\def\longleftarrow{\leftarrow\joinrel\rightarrow}
```

Symbol ten powstał z połączenia dwóch strzałek \leftarrow i \rightarrow . A oto inne efekty:

```

 $\overset{a}{b}$  \{a\atop b\}      $\overset{a}{\underset{b}{}}$  \{a\above2pt b\}

```

```

 $\{a\}$  \{a\brace b\}      $\binom{a}{b}$  \{a\choose b\}      $\lceil a \rceil$  \{a\brack b\}

```

Pierwszy jest przydatny w składzie podwójnych granic całek, sum, itd., drugi w wyrażeniach z uławkami piętrowymi, a pozostałe trzy mają wiele zastosowań, np. w macierzach dwuwierszowych w tekście. W szczególnych przypadkach można zastosować komendę `\abovewidthdelims`, pozwalającą wyraźnie określić ograniczniki i wyspecyfikować grubość kreski ułamka. Po zdefiniowaniu komendy:

```
\def\oddiy{\abovewidthdelims\lbrace\lbrace1pt}
```

można złożyć wyrażenie postaci „ $\overset{a}{b}$ ” (`\oddiy b`).

I na koniec kilka przydatnych komend (główną jest `\stacksymbols`):

```

\def\stacksymbols #1#2#3#4{\def\theguybelow{#2}
\def\verticalposition{\lower#3pt}
\def\spacingwithinsymbol{\baselineskip0pt\lineskip#4pt}
\mathrel{\mathpalette\intermediary#1}}
\def\intermediary#1#2{\verticalposition\box{\spacingwithinsymbol
\everycr={}\tabskip0pt
\halign{\$ \mathsurround0pt#1\hfil#\hfil\$ \crr#2\crr
\theguybelow\crr}}}

```

```
\def\lapproxeq{\stacksymbols{<}{\sim}{2,5}{2}}
```

```
\def\gapproxeq{\stacksymbols{>}{\sim}{2,5}{2}}
```

matematyka:
funkcje specjalne

Ostatnie dwa wiersze definiują nowe symbole \lesssim i \gtrsim , omówione pod hasłem **matematyka: relacje**. Ich definicje oparte są na nowej komendzie `\stacksymbols`. Symbole te automatycznie zmniejszają swój rozmiar, jeśli są składane jako indeksy. `\stacksymbols` pozwala konstruować wiele złożonych symboli, jak również dowolnie regulować odstępy. Na przykład

$$a \quad \underset{\sim}{g} \quad b$$

otrzymano z następującego pliku wejściowego:

```

 $\$a\qquad$ 
 $\stacksymbols{g}{\tilde{}}{8}{1}\qquad$ 
 $\stacksymbols{g}{\tilde{}}{-1}{4}\qquad$ 
 $b\$\$$ 

```

Pierwsze dwa argumenty komendy `\stacksymbols` to symbole, które pojawią się jeden nad drugim, następne kontrolują ich pozycję w pionie oraz odstęp między nimi.

W definicji komendy `\stacksymbols` wykorzystano dwie komendy nie omówione w podręczniku. `\mathrel` tworzy nowe symbole relacji — \TeX wówczas stosuje zasady odstępów właściwe dla relacji. `\mathpalette` (działająca na dwa argumenty) zapewnia elastyczność nowego symbolu w zależności od miejsca jego wystąpienia.

matematyka: funkcje specjalne 38–43

Funkcje typu \sin , \cos czy \log , jak również kilka innych wyrażeń zwykle składa się nawet we wzorach prostym krojem pisma. W \TeX -u uzyskuje się je za pomocą komend:

```

 $\arc\cos$   $\arc\sin$   $\arctg$   $\arg$   $\cos$   $\cosh$   $\ctg$   $\ctgh$ 
 $\csc$   $\deg$   $\det$   $\dim$   $\exp$   $\gcd$   $\hom$   $\inf$ 
 $\ker$   $\lg$   $\lim$   $\liminf$   $\limsup$   $\ln$   $\log$   $\max$ 
 $\min$   $\Pr$   $\sec$   $\sin$   $\sinh$   $\sup$   $\tg$   $\tgh$ 

```

Możliwe jest umieszczenie dowolnego wyrażenia nad tymi symbolami w wyrażeniach eksponowanych. Na przykład

$$\lim_{x \rightarrow \infty} \quad \overset{\heartsuit}{\max}$$

Zwykle nie stosuje się ich we frakcji górnej wzorów. Tylko w przypadku niektórych symboli frakcje górna czy dolna są umieszczane nad lub pod symbolem (`\det`, `\gcd`, `\inf`, `\lim`, `\liminf`, `\limsup`, `\max`, `\min`, `\Pr`, `\sup`). W pozostałych przypadkach \TeX umieszcza indeksy w normalnym położeniu:

$$\sin^2 x + \cos^{-2} x \neq 1.$$

Plik wejściowy ostatniego fragmentu ma postać:

```

 $\$\sin^2x + \cos^{-2}x \neq 1.\$\$$ 

```

Istnieją także dwie odmiany symbolu \bmod , w zależności od tego, czy występuje on pomiędzy zmiennymi, czy na końcu wzoru; są to, odpowiednio, `\bmod` i `\pmod`. A oto przykład ich działania: `\(a\bmod b)` składa $(a \bmod b)$, a wpis `\pmod{n}` daje w efekcie $(\bmod n)$.

matematyka: interpunkcja 24–33

W trybie matematycznym można wykorzystywać tradycyjne znaki interpunkcyjne, przy czym wpis np. `$a:b$` daje skład: $a : b$ (odstępny są właściwe dla relacji). Aby złożyć dwukropek z odstępami właściwymi dla interpunkcji, należy zastosować komendę `\colon`; np. `$a\colon b$` składa $a : b$.

W wyrażeniach matematycznych składanych w tekście (wzór nie eksponowany) znaki interpunkcji należy umieszczać poza trybem matematycznym. Aby otrzymać w składzie wyrażenie $a = x, y$ lub z , należy w pliku wejściowym umieścić wpis `$a=x$, y lub z` (nie zaś np. `$a=x, y$ lub z`), co pozwoli na ujednoczenie odstępów i podział wiersza. Aby wiersz nie był dzielony w danym miejscu, należałoby umieścić w tekście wiązanie (tyldę `~`; patrz podział wierszy).

W wyrażeniach eksponowanych (między podwójnymi znakami `$`) interpunkcję umieszcza się jako część równania, chyba że w skład wyrażenia wchodzi tekst. Wówczas tekst i związana z nim interpunkcja powinny być umieszczone w pudełku `\hbox` (patrz równanie 13 w przykładzie IX).

matematyka: odstęp 14–15, 38–43, 60–63

\TeX różnicuje odstęp w zależności od tego, czy pojawiają się one między znakami operacji, relacji, interpunkcji, po lewym czy prawym ograniczniku, itd. Za pomocą komend:

$$\!, \!, \!>, \!>, \quad \text{oraz} \quad \!>>, \!>>$$

których efekt działania przedstawiony został we wstępie (pierwsza tworzy wąski odstęp ujemny, pozostałe są odstępami dodatnimi różnej długości), można dowolnie przearanżować te odstęp. Definicje pierwszych czterech przedstawione są pod hasłem słowa kluczowe; `\quad` i `\qquad` są zdefiniowane w osobnych opisach.

Można „oszukać” \TeX -a, używając symboli grupujących; np. `a,b` składa a, b , natomiast `$a{,}b$` składa a, b ; z kolei `$a:b$` składa $a : b$, natomiast `$a{:}b$` składa $a : b$. Inne sposoby kontroli odstępów poziomych przedstawiono w pliku wejściowym do równania 8 w przykładzie IX, a także pod hasłem `\mskip`.

W równaniach 5 i 10 (patrz przykład IX) pokazano sposób kontroli odstępów pionowych za pomocą odmiany komendy `\phantom`. `\vphantom` w połączeniu ze `\smash` pozwala dowolnie umieszczać symbole; `\phantom` może określać położenie indeksów. Na przykład

$$R_{abc}^d \text{ otrzymano z wpisu } \$R_{\{abc\}^{d}}$,

$$R_{cd}^{ab} \text{ otrzymano z wpisu } \$R_{cd}^{\{ab\}}$.$$$$

Te same efekty można uzyskać w dużo prostszy sposób, „oszukując” \TeX -a; np. plik wejściowy dla:

$$R_{cd}^{ab}, \quad R_d^{abc} \quad \text{oraz} \quad R_b^a c_d$$

ma postać:

$$\$R^{\{ab\}}_{\{cd\}}, \quad \!>> R^{\{abc\}}_{\{d\}} \quad \!>> \!>> R^a_{\{b\}} c_{\{d\}} \!>> \$.$$

Należy podkreślić, że pionowe rozmieszczenie indeksów w stylu eksponowanym jest trochę inne niż w stylu tekstowym. Rozmieszczenie to można kontrolować za pomocą komend `\raise` i `\lower` (patrz równanie 5 w przykładzie IX oraz położenie tekstu na stronie).

Położenie akcentów nad akcentami także wymaga dopasowania. W takim przypadku wykorzystuje się komendę `\skew`. „Ukos” można kontrolować liczbowo; np. plik wejściowy symboli \hat{I} , \hat{I} , \hat{I} i \hat{I} ma postać:

matematyka:
ograniczniki

... symboli $\hat{\hat{I}}$, $\skew3\hat{\hat{I}}$, $\skew5\hat{\hat{I}}$ i $\skew9\hat{\hat{I}}$ ma postać...

(patrz także `\rlap`, `\llap`, matematyka: interpunkcja oraz `\mathsurround`).

matematyka: ograniczniki 42–43, 54–55, 60–63

Pewne symbole, takie jak klamry, nawiasy kwadratowe, nawiasy okrągłe, określają, gdzie wyrażenie zaczyna się i gdzie kończy.

<code>(</code>	<code>(\$</code>	<code>)</code>	<code>)\$</code>	<code>[</code>	<code>[\$</code>	<code>]</code>	<code>]\$</code>
<code>{</code>	<code>{\$</code>	<code>}</code>	<code>}\$</code>	<code> </code>	<code> \$</code>	<code> </code>	<code> \$</code>
<code>⌊</code>	<code>\lfloor</code>	<code>⌋</code>	<code>\rfloor</code>	<code>⌈</code>	<code>\lceil</code>	<code>⌉</code>	<code>\rceil</code>
<code>\</code>	<code>\backslash</code>	<code>/</code>	<code>/\$</code>	<code><</code>	<code>\langle</code>	<code>></code>	<code>\rangle</code>

Nawiasy `{ i }` można uzyskać za pomocą komend `\lbrace` i `\rbrace`, nawiasy `[i]` — za pomocą `\lbrack`, `\rbrack`, a kreski `| i ||` — za pomocą `\vert` i `\Vert`. Pionowe strzałki wymienione pod hasłem matematyka: relacje mogą również być wykorzystane jako ograniczniki. Wszystkie te ograniczniki automatycznie zwiększają swój rozmiar wraz z wyrażeniem, jeśli są poprzedzone komendą, odpowiednio, `\left` i `\right` (patrz przykład IX oraz osobne wpisy dla `\right` i `\left` w dodatku). Z wyjątkiem ograniczników z ostatniego wiersza w zestawieniu, wszystkie pozostałe mogą powiększać swoje rozmiary.

Użytkownik może również wprost określić kilka dużych rozmiarów ograniczników, poprzedzając odpowiednią komendę przedrostkiem `\big`, `\Big`, `\bigg` lub `\Bigg`. Efekt działania takich kombinacji jest następujący:

| | | | |

Pierwsza kreska ma „normalny rozmiar”. Te komendy kontrolujące rozmiar ograniczników mogą być użyte pojedynczo, podczas gdy `\left` i `\right` muszą występować parami w danym równaniu. Można także określić, czy dany duży ogranicznik występuje z lewej czy z prawej strony wyrażenia, czy w jego środku, poprzez wpis „`\bigl`”, „`\bigr`” czy „`\bigm`”. Jest to ważne ze względu na odstępy w równaniu, ponieważ TeX zostawia różne odstępy po ograniczniku, w zależności od miejsca jego wystąpienia. Oto rozmiary `\Bigg` wszystkich ograniczników:

() [] { } [] [] < > / \ | || ↑ ↓ ⇕ ⇓ ⇔ ⇔

I na koniec kilka ekstra ograniczników; niektóre z nich występują tylko w dużych rozmiarach; ich minimalny rozmiar jest następujący:

| | || || | | () ∫ }

Składające je komendy to, kolejno, `\arrowvert`, `\Arrowvert` i `\bracevert`; `\lgroup` i `\rgroup`; `\lmoustache` i `\rmoustache`. Są one częścią innych symboli i powinny być poprzedzone komendami `\left`, `\right` czy `\big`. Na przykład plik wejściowy ostatnio eksponowanej części ma postać:

```
...
\left\lgroup{\quad}\right\rgroup\quad
\left\lmoustache{\quad}\right\rmoustache$$
```


matematyka: operacje 24–33, 52–57, 60–63

$+$ <code>+\$</code>	\ddagger <code>\$\$\ddagger\$</code>	\dagger <code>\$\$\dagger\$</code>	\odot <code>\$\$\odot\$</code>
$-$ <code>-\$</code>	\setminus <code>\$\$\setminus\$</code>	\cdot <code>\$\$\cdot\$</code>	\oplus <code>\$\$\oplus\$</code>
$/$ <code>/\$</code>	\times <code>\$\$\times\$</code>	\star <code>\$\$\star\$</code>	\ominus <code>\$\$\ominus\$</code>
\cdot <code>\$\$\cdot\$</code>	\cap <code>\$\$\cap\$</code>	\cup <code>\$\$\cup\$</code>	\otimes <code>\$\$\otimes\$</code>
\pm <code>\$\$\pm\$</code>	\sqcap <code>\$\$\sqcap\$</code>	\sqcup <code>\$\$\sqcup\$</code>	\oslash <code>\$\$\oslash\$</code>
\mp <code>\$\$\mp\$</code>	\triangleleft <code>\$\$\triangleleft\$</code>	\bullet <code>\$\$\bullet\$</code>	\bigcirc <code>\$\$\bigcirc\$</code>
\wr <code>\$\$\wr\$</code>	\triangleright <code>\$\$\triangleright\$</code>	\uplus <code>\$\$\uplus\$</code>	\circ <code>\$\$\circ\$</code>
\div <code>\$\$\div\$</code>	\triangleup <code>\$\$\triangleup\$</code>	\vee <code>\$\$\vee\$</code>	\diamond <code>\$\$\diamond\$</code>
$*$ <code>\$\$\ast\$</code>	\triangledown <code>\$\$\triangledown\$</code>	\wedge <code>\$\$\wedge\$</code>	\amalg <code>\$\$\amalg\$</code>

Komenda `$$\setminus$` składa znak `\`, podobnie jak komenda `$$\backslash$` (wymieniona poniżej pod hasłem matematyka: operacje), jednak w obu przypadkach są inne zasady dotyczące odstępów: np. wpis `$$A\setminus B$` składa wyrażenie $A \setminus B$, podczas gdy `$$A \backslash B$` — wyrażenie $A \setminus B$. Istnieją też alternatywne komendy dla symboli `$$\wedge$` i `$$\vee$`: `$$\land$` i `$$\lor$`. Podobnie `$$\not$` jest alternatywą dla `$$\neg$` (wymienione one są pod hasłem matematyka: symbole). Niektóre operatory występują w dużych rozmiarach. Większy element każdej z wymienionych poniżej par jest automatycznie wykorzystywany w równaniach eksponowanych.

\sum <code>\$\$\sum\$</code>	\bigodot <code>\$\$\bigodot\$</code>	\bigotimes <code>\$\$\bigotimes\$</code>
\bigoplus <code>\$\$\bigoplus\$</code>	\bigcap <code>\$\$\bigcap\$</code>	\bigcup <code>\$\$\bigcup\$</code>
\bigsqcup <code>\$\$\bigsqcup\$</code>	\biguplus <code>\$\$\biguplus\$</code>	\prod <code>\$\$\prod\$</code>
\coprod <code>\$\$\coprod\$</code>	\int <code>\$\$\int\$</code>	\oint <code>\$\$\oint\$</code>
\bigvee <code>\$\$\bigvee\$</code>	\bigwedge <code>\$\$\bigwedge\$</code>	

Dostępny jest także mały znak całki, \int , składany za pomocą komendy `$$\smallint$`. Granice można określać dla całek, sum, iloczynów. Są one wprowadzane jako frakcja górna i dolna. W niektórych przypadkach granice automatycznie pojawiają się nad i pod symbolem; w innych umieszczane są z prawej strony symbolu. Rozmieszczenie to może być zmienione za pomocą komend `\limits` (granice nad lub pod symbolem) i `\nolimits` (granice z boku symbolu). Patrz przykłady VII i IX.

matematyka: podział wierszy 14

Wyrażenia matematyczne w tekście (umieszczone pomiędzy pojedynczymi znakami `$`) mogą być dzielone tylko w kilku miejscach: po znakach relacji, jak `=` czy `<`, lub (mniej preferowane miejsca) po symbolach operacji, jak `+` czy `-`. We wszystkich tych przypadkach podział wiersza nastąpi tylko wtedy, gdy symbol nie jest schowany wewnątrz wyrażnej grupy (nie leży pomiędzy nawiasami `{, }`). Tak więc przenoszenie wierszy może być zabronione w ukrytych częściach wyrażenia lub w całym wyrażeniu, jeśli otoczone zostanie ono klamrami — należy więc uważać, by nie ingerować w zasady odstępów w wyrażeniach matematycznych (patrz matematyka: odstępy). Podział wierszy może być dozwolony w miejscach nie wybranych przez $\text{T}_{\text{E}}\text{X}$ -a — w wyniku zastosowania komendy `\allowbreak`.

W przypadku iloczynów podział wiersza może być sugerowany poprzez umieszczenie w miejscach kolejnych znaków mnożenia komend `*`, np. $x = (a - b)(c - d) \times (u - v)(x - y)$. Plik wejściowy tego wyrażenia ma postać:

```
... $x=(a-b)\*(c-d)\*(u-v)\*(x-y)$ ...
```


Jak łatwo zauważyć, \TeX ignoruje komendę `*`, chyba że jest to dobre miejsce do podziału wiersza, wtedy wstawia tam znak \times .

Wyrażenia eksponowane, umieszczone między podwójnymi znakami $\$$, nie są nigdy automatycznie przenoszone między wierszami, należy dzielić je ręcznie. W wielu sytuacjach lepiej jest eksponować układy równań czy wyrażenia wielowierszowe za pomocą komendy `\displaylines` (patrz przykład IX), niż umieszczać w każdym wierszu podwójne znaki $\$$ (co skutkuje zbyt dużymi odstępami między równaniami).

matematyka: relacje 24–33, 38–43, 52–57, 60–63

$<$	<code>\<</code>	$>$	<code>\></code>	$=$	<code>\=</code>
\perp	<code>\perp</code>	\leq	<code>\leq</code>	\geq	<code>\geq</code>
\equiv	<code>\equiv</code>	$\dot{=}$	<code>\doteq</code>	\prec	<code>\prec</code>
\preceq	<code>\preceq</code>	\succ	<code>\succ</code>	\succeq	<code>\succeq</code>
\sim	<code>\sim</code>	\simeq	<code>\simeq</code>	\ll	<code>\ll</code>
\gg	<code>\gg</code>	\subset	<code>\subset</code>	\subseteq	<code>\subseteq</code>
\supset	<code>\supset</code>	\supseteq	<code>\supseteq</code>	\sqsubseteq	<code>\sqsubseteq</code>
\sqsupseteq	<code>\sqsupseteq</code>	\asymp	<code>\asymp</code>	\approx	<code>\approx</code>
\cong	<code>\cong</code>	\bowtie	<code>\bowtie</code>	\models	<code>\models</code>
\vdash	<code>\vdash</code>	\dashv	<code>\dashv</code>	\smile	<code>\smile</code>
\frown	<code>\frown</code>	\mid	<code>\mid</code>	\parallel	<code>\parallel</code>
\in	<code>\in</code>	\ni	<code>\ni</code>	\propto	<code>\propto</code>

Relacje te mogą być negowane za pomocą komendy `\not`, np. `\not\perp` składa \nperp . Nie zawsze daje to oczekiwane rezultaty: `\not\parallel` tworzy znak \nparallel . W niektórych przypadkach już są zdefiniowane zaprzeczenia: `\notin` daje \notin , `\neq` lub `\ne` składa znak \neq (w niniejszym dodatku komenda `\ne` została przedefiniowana). Czasem samemu można poprawić niektóre symbole za pomocą komendy `\kern`: pisząc `\not\kern1,4pt\parallel` otrzymamy \nparallel . Istnieją także komendy działające zamiennie, np. `\le` i `\ge` dają taki sam efekt, jak `\leq` i `\geq`; `\owns` jest równoważne `\ni`.

Inne dostępne w \TeX -u symbole relacji to:

\leftarrow	<code>\leftarrow</code>	\longleftarrow	<code>\longleftarrow</code>
\rightarrow	<code>\rightarrow</code>	\longrightarrow	<code>\longrightarrow</code>
\Lleftarrow	<code>\Lleftarrow</code>	\Longleftarrow	<code>\Longleftarrow</code>
\Rrightarrow	<code>\Rrightarrow</code>	\Longrightarrow	<code>\Longrightarrow</code>
\uparrow	<code>\uparrow</code>	\downarrow	<code>\downarrow</code>
\Uparrow	<code>\Uparrow</code>	\Downarrow	<code>\Downarrow</code>
\leftrightarrow	<code>\leftrightarrow</code>	\longleftrightarrow	<code>\longleftrightarrow</code>
\Lleftrightarrow	<code>\Lleftrightarrow</code>	\Longleftrightarrow	<code>\Longleftrightarrow</code>
\updownarrow	<code>\updownarrow</code>	\Updownarrow	<code>\Updownarrow</code>
\mapsto	<code>\mapsto</code>	\longmapsto	<code>\longmapsto</code>
\hookrightarrow	<code>\hookrightarrow</code>	\hookleftarrow	<code>\hookleftarrow</code>
\nearrow	<code>\nearrow</code>	\searrow	<code>\searrow</code>
\swarrow	<code>\swarrow</code>	\nwarrow	<code>\nwarrow</code>
\leftharpoonup	<code>\leftharpoonup</code>	\rightharpoonup	<code>\rightharpoonup</code>
\leftharpoondown	<code>\leftharpoondown</code>	\rightharpoondown	<code>\rightharpoondown</code>
\rightleftharpoons	<code>\rightleftharpoons</code>		

Wszystkie strzałki pionowe występują także w większych rozmiarach; uzyskuje się je za pomocą tych samych komend co w przypadku ograniczników. Alternatywnymi komendami `\rightarrow` i `\leftarrow` są `\to` i `\gets`. Komenda `\iff`

składa znak \iff , podobnie jak \Leftrightarrow , z tym że z większym odstępem z jednej strony.

I na koniec dwa symbole uzyskiwane w Plain TeX-u dopiero po ich zdefiniowaniu (patrz matematyka: efekty specjalne):

\lapprox składa \lesssim , \gapprox zaś składa \gtrsim .

matematyka: stopnie i odmiany pisma 40–41, 60–63

Standardowy krój pisma używany w wyrażeniach matematycznych nazywany jest *kursywą matematyczną*. W TeX-u istnieją również inne kroje pisma, właściwe symbolom matematycznym. Występują one w różnych odmianach, różniących się stopniami. Są to: styl tekstowy (*textstyle* — do zwykłych wyrażień wewnątrz tekstu), styl eksponowany (*displaystyle* — do wyrażień eksponowanych w osobnym wierszu) oraz styl wskaźnikowy (*scriptstyle* i *scriptscriptstyle* — do indeksów). TeX sam dobiera odpowiedni styl do poszczególnych części wyrażenia. Użytkownik może także za pomocą odpowiednich instrukcji wybrać potrzebny mu styl $\texttt{\textstyle}$, $\texttt{\scriptstyle}$, itd. (patrz równanie 1 w przykładzie V). Ma także do dyspozycji kaligrafowany krój pisma (patrz $\texttt{\cal}$). Pakiety bardziej zaawansowane niż Plain TeX, specjalnie przystosowane do składu wyrażenia matematycznych (np. $\mathcal{M}\mathcal{S}\text{-TeX}$, pakiet stworzony przez American Mathematical Society), oferują jeszcze inne odmiany krojów pisma.

Logo $\mathcal{M}\mathcal{S}\text{-TeX}$ zostało złożone komendą:

```
 $\cal A\kern-,1667em\lower,5ex\hbox{\cal M}\kern-,075em S\text{-TeX}$ 
```

matematyka: symbole 32–33, 38–43, 52–57, 60–63

\aleph	\forall	\exists
\prime	\hbar	\emptyset
∇	\neg	\surd
\imath	\jmath	ℓ
\flat	\sharp	\natural
\top	\perp	\backslash
\Re	\Im	∂
∞	\angle	\triangle
\clubsuit	\diamondsuit	\heartsuit
\spadesuit	\dots	\cdots
\wp	\cdot	\cdot
\vdots	\ddots	

matematyka: wyrażenia eksponowane 24–31, 38–43, 52–57, 60–63

Wyrażenia eksponowane to takie, które znajdują się w osobnym wierszu. Są one automatycznie wyśrodkowane, z odstępem nad i pod wyrażeniem. Odstępy między składnikami takiego wyrażenia są trochę inne niż we wzorach w tekście (patrz matematyka: odstępy), ponadto niektóre wykorzystywane stopnie pisma są większe niż w trybie tekstowym. Styl ten jest zwany stylem eksponowanym, może być wywołany w tekście (bez eksponowania) za pomocą komendy $\texttt{\displaystyle}$.

Odstęp nad i pod eksponowanym równaniem można zwiększyć przez dodanie dowolnego dodatkowego odstępu pionowego (za pomocą komend $\texttt{\smallskip}$, $\texttt{\medskip}$, itd., lub przez pozostawienie w pliku wejściowym dodatkowego pustego wiersza). W zależności od typu wyrażenia eksponowanego można chcieć lub nie tego

`\mathstrut`

odstępu. Cztery parametry, których wartości można zmieniać, kontrolują odstępy nad i pod wyrażeniami eksponowanymi. Te parametry, wraz z ich wartościami domyślnymi w Plain TeX-u, to:

```
\abovedisplayskip=12pt plus 3pt minus 9pt
\abovedisplayshortskip=0pt plus 3pt
\belowdisplayskip=12pt plus 3pt minus 9pt
\belowdisplayshortskip=7pt plus 3pt minus 4pt
```

Pierwsza i trzecia komenda określają, odpowiednio, odstępy nad i pod wyrażeniem eksponowanym. Druga i czwarta określają odstępy wstawiane, gdy wiersz bezpośrednio poprzedzający wyrażenie eksponowane jest krótki. We wszystkich przypadkach **plus** i **minus** odnoszą się do elastyczności odstępu.

`\mathstrut`

Składa w wyrażeniach matematycznych niewidoczną kreskę o zerowej szerokości, a wysokości i głębokości odpowiadającej znakowi nawiasu (, co pozwala na ujednoczenie rozmiaru różnych wyrażień. Na przykład plik wejściowy wzorów

$$\sqrt{a} + \sqrt{b} \quad \text{i} \quad \sqrt{a} + \sqrt{b}$$

ma postać:

```
$$\sqrt{a} + \sqrt{b} \quad \text{i} \quad \sqrt{a} + \sqrt{b}
\mathstrut a + \mathstrut b$$
```

Komenda `\mathstrut` jest zdefiniowana w Plain TeX-u jako `\vphantom` (i wbrew nazwie może także być wykorzystana poza trybem matematycznym).

`\mathsurround` *

Parametr określający, jak duży odstęp ma znaleźć się w tekście przed i za wyrażeniem matematycznym. Wartość tego parametru w Plain TeX-u wynosi **0pt**. Oto efekt działania dwóch różnych wartości parametru `\mathsurround`:

2 punkty: $a < b$ i $b < c \Rightarrow a < c$.

5 punktów: $a < b$ i $b < c \Rightarrow a < c$.

Plik wejściowy tego fragmentu ma postać:

```
{\mathsurround=2pt 2 punkty: $a<b$ i $b<c$ $\Rightarrow$ $a<c$.}
\hfil\break
{\mathsurround=5pt 5 punktów: $a<b$ i $b<c$ $\Rightarrow$ $a<c$.}
```

`\medbreak`

Sugeruje złamanie kolumny lub umieszczenie odstępu o rozmiarze `\medskip`.

`\medskip` x-xiii, 10–13, 66–67

Wstawia odstęp pionowy $\rule{0pt}{1em}$. Domyślny rozmiar tego odstępu, zwany `\medskipamount`, wynosi **6pt plus2pt minus2pt**; tj. odstęp 6-punktowy, który może się skurczyć lub rozciągnąć, jeśli to konieczne. `\medskip` jest zdefiniowana jako `\vskip\medskipamount` (patrz `\bigskip`).

METAFONT

Program do tworzenia fontów, napisany przez Donalda Knutha.

`\midinsert` 46–49

Próbuje wstawić materiał w miejscu wystąpienia komendy lub u góry następnej wolnej strony (patrz przykład VI).

`minus` 11, 50

Określa elastyczność kleju; np. komenda `\hskip 10pt minus 4pt` złoży odstęp o rozmiarze pomiędzy 6 a 10 punktów (patrz plus i słowa kluczowe).

`\month` *

Na początku pracy \TeX sprawdza datę w komputerze. Za pomocą komendy `\month` można uzyskać informację dotyczącą aktualnej daty miesięcznej. Teraz mamy miesiąc 3. Plik wejściowy ostatniego wiersza ma postać:

```
Teraz mamy miesiąc \number\month. Plik ...
```

Cyfry są dobre w odniesieniu do dni w roku czy samych lat, ale głupio brzmi powiedzenie „Moje urodziny są w miesiącu 8, zacznij zbierać pieniądze na prezent w miesiącu 5, dobrze?” Oto nowa komenda dokonująca konwersji cyfr na nazwy. Pozwoli ona na komentarze typu: „Jest marzec; jak przyjemnie jest w Bombaju o tej porze roku”. Plik wejściowy tego fragmentu ma postać:

```
\def\monthname {\ifcase\month\or styczeń\or luty\or marzec\or
kwiecień\or maj\or czerwiec\or lipiec\or sierpień\or wrzesień\or
październik\or listopad\or grudzień\fi}
\def\Bombayclimate {\ifcase\month\or bardzo zimno\or zimno\or przyjemnie\or
ciepło\or gorąco\or sucho\or mokro\or bardzo mokro\or miło\or bardzo
ciepło\or miło\or fatalnie\fi}
Pozwoli ona na komentarze typu: „Jest \monthname; jak
\Bombayclimate jest w Bombaju o~tej porze roku. Plik ...
```

`\moveleft` * 74–75

Przesuwa pudełko w lewo. Działa wyłącznie w trybie pionowym. Patrz tryby pracy.

`\moveright` * 74–75

Przesuwa pudełko w prawo. Patrz `\moveleft`.

`\mskip` *

Komenda z trybu matematycznego, analogiczna do `\hskip`. Żądany odstęp musi być wyspecyfikowany w matematycznych jednostkach `mu` (patrz słowa kluczowe). `\hskip` również działa wewnątrz wzorów, jednak nie może być wyspecyfikowana w jednostkach `mu`; częściej preferowaną komendą jest `\mskip` (automatycznie dopasowuje się do rozmiaru pisma obowiązującego w danym wzorze).

`\multiply` * 2, 72–73, 78–79, 80–81

Mnoży wartość licznika przez liczbę całkowitą (patrz przykład XIV).

`\multispan` 44–45, 48–49

Wykorzystana do łączenia wielu kolumn w tabeli (patrz tytuł tabeli 2 lub podpis pod tabelą 3 w przykładzie VI).

nakładanie

Patrz `\rlap` i `\llap`.

`\narrower``\narrower` vi–vii, 6–7, 8, 20–21, 30–31

Eksponuje tekst w formacie węższym o bieżące wcięcie akapitu (wartość parametru `\parindent`). Komenda ta jest zdefiniowana w Plain T_EX-u za pomocą `\rightskip` i `\leftskip` (patrz akapity).

`\negthinspace` 10–11

Ścisła tekst w następujący sposób. Efekt taki można uzyskać wpisując w pliku wejściowym „`w\negthinspace następujący\negthinspace sposób`”. Komenda ta jest zdefiniowana w Plain T_EX-u jako:

```
\def\negthinspace{\kern-,16667em}
```

`\newbox` 80

Przydziela rejestry `\box`, tak jak `\newcount` przydziela rejestry `\count`. Komenda ta jest przydatna z takich samych powodów jak dotyczące `\newcount`. Kiedy rejestr jest już przydzielony (oznaczony), komenda `\box` działa trochę inaczej niż `\count`. Na przykład *tekst* powstał z pliku wejściowego:

```
\newbox\justtesting \setbox\justtesting=\hbox{\it tekst\}%
\box\justtesting powstał z pliku wejściowego:
```

(patrz `\box`).

`\newcount` 68–73, 78–81, 88–89

T_EX oznacza rejestry wewnętrzne za pomocą nieujemnych liczb całkowitych: np. rejestry `\count` są oznaczone `\count0`, `\count1`, ..., `\count255`. Ponieważ ciężko jest zapamiętać, który rejestr zawiera którą zmienną, można nazwać rejestry za pomocą komendy `\newcount` (T_EX automatycznie przypisze do tej nazwy numer rejestru jeszcze nie wykorzystany). Wtedy można odwoływać się do danego rejestru na podstawie jego nazwy. Na przykład `\newcount\var` oznacza rejestr `\count`, do którego odtąd będzie można odwoływać się poprzez nazwę `\var` (patrz `\count`).

Pomysł wykorzystania `\newcount` (tak jak i innych komend `\new...`) trzeba stosować z umiarem. W celu tymczasowego przechowania informacji należy wykorzystywać któryś z rejestrów pomocniczych (*scratch*). Także wewnątrz grupy można wykorzystywać dowolny z rejestrów. `\newcount` można stosować do własnych oznaczeń rejestrów, które będą wykorzystane w wielu miejscach w dokumencie.

`\newdimen` 80

Komenda podobna do `\newcount`, z tym że zapisuje w rejestrach wymiary (patrz `\dimen`).

niepełne pudełka

Patrz błędy oraz pudełka.

`\noalign` * 40–41, 48–49, 54–57, 60–63

Pozwala na umieszczenie wewnątrz wyrównania niewyrównanego materiału.

`\nobreak` 66–67

Zabezpiecza przed podziałem wiersza przy poziomym przetwarzaniu tekstu lub przed złamaniem strony w pionowym trybie pracy (patrz tryby pracy).

`\noindent` * vi–xiii, 6–9, 58–59, 66–67

Likwiduje wcięcie akapitu.

`$$\nolimits$` * 52, 60–61

Patrz matematyka: operacje.

`\nopagenumbers` 2

Likwiduje trwale numerację stron.

nowe komendy w podręczniku

Przy składzie tej książki zdefiniowano kilka nowych komend. Wiele z nich to skróty, wykorzystywane czysto lokalnie (np. w kilku kolejnych wierszach); inne określają kroje pisma lub są komendami specyficznymi dla formatu tej książki. Oprócz instrukcji `\title` (odnoszącej się do dużego kroju pisma w tytułach rozdziałów) i komendy `\newpageno` (umieszczającej na marginesie pogrubione numery stron), żadne nowe komendy nie były wykorzystywane aż do przykładu X. Pozwala to użytkownikowi na bezpośrednie skopiowanie i wykorzystywanie tej części pliku wejściowego podręcznika.

W pozostałej części książki, w większości przykładów X–XVII, są zdefiniowane nowe instrukcje. Jeśli plik wejściowy zawiera definicję nowej komendy, nie można jej pominąć przy kopiowaniu. Strony wejściowe książki, dodatek i uwagi końcowe zostały sformatowane przy użyciu dużej liczby specjalnie stworzonych instrukcji. Komendy te przedstawione są w uwagach końcowych.

A oto lista głównych komend zdefiniowanych w tej książce:

`\letterhead`, `\date`, `\address`, `\sal` i `\endletter`: 66.

`\memo`: 68.

`\pattern`: 68.

`\beginscript` i `\endscript`: 70.

`\pow` i `\fermat`: 72.

`\frame`, `\fitframe`, `\shframe` i `\s`: 76.

`\trap` i `\rect`: 78.

`\beginart` i `\endart`: 80.

`\chartable`: patrz kody znaków.

`\ignore`: patrz komentarze.

`\monthname`: patrz `\month`.

`\stacksymbols`, `\lapprox` i `\gapprorx`: patrz matematyka: efekty specjalne.

`\table`, `\caption` i komendy związane: patrz tabele.

`\now` i `\today`: patrz `\year`.

`\raggedleft`, `\beginliteral` i `\endliteral`: patrz uwagi końcowe.

`\null`

Skrót komendy `\hbox{}`. Komendę tę wykorzystuje się w celu „oszukania” T_EX-a. Będzie on wówczas honorował komendy dotyczące odstępów, które są normalnie ignorowane na początku danego wiersza czy strony. Na przykład `\null\skip` pozostawi odstęp w sytuacji, gdy komenda `\hskip` nie będzie działać; podobnie `\null\vskip` czy `\null\vfil` wstawi odstęp u góry nowej strony.

`\number``\number * 72–73`

Podaje wartość licznika (patrz `\day`, `\month` i `\year`).

numerowanie równań

Patrz przykłady II, V, VII i IX, oraz `\eqno`, `\leqno`, `\eqalignno` i `\leqalignno`.

numerowanie stron vi–xiii, 2, 80–81

- *Informacje ogólne.* Numery stron są automatycznie środkowane u dołu strony. Można wyłączyć opcję numerowania stron za pomocą komendy `\nopagenumbers`; można także zmienić numer pierwszej strony za pomocą komendy `\pageno`, tak jak to zrobiono na początku przykładu IX. Numerację rzymską stron uzyskuje się wstawiając ujemną wartość `\pageno` (patrz str. vi i viii).
- *Szczegóły* (patrz `\pageno`). Automatyczna numeracja stron w Plain TeX-u osiągnięta jest wewnętrznie za pomocą komendy `\folio`. Komenda ta pobiera aktualny numer strony z `\pageno`; jeśli wartość ta jest ujemna, `\folio` dokonuje konwersji cyfry na rzymską (za pomocą instrukcji `\romannumeral`). Domyślnie zdefiniowana stopka w Plain TeX-u zawiera wartość `\folio`, złożoną 10-punktową antykwą (patrz `\folio` i `\footline`). Komenda `\nopagenumbers` powoduje wewnętrzne przededefiniowanie stopki: `\footline={\hfil}`. Można także uzyskać inne efekty przeddefiniując `footline` (lub `headline`, gdy numery mają pojawić się u góry strony, jak na str. 2). Komendy warunkowe (włączając `\ifnum`) są przydatne do zmiany położenia numerów stron, do zaniechania numeracji kilku stron lub do ustawiania typów nagłówek czy stopek na głównych stronach (patrz definicja `\footline` na początku przykładu XVII oraz definicja komendy `headline` w uwagach końcowych).

\obeylines

Ta komenda powoduje, że TeX składa materiał zachowując układ wierszy z pliku wejściowego, tj. każda linia tekstu z pliku wejściowego (o ile nie jest za długa) będzie znajdować się w osobnym wierszu. W efekcie każda linia pliku wejściowego jest traktowana jako osobny akapit.

(Na przykład plik wejściowy tego fragmentu ma postać:)

```
{\obeylines
Ta komenda powoduje, że TeX\ składa materiał
zachowując układ wierszy z pliku wejściowego, tj.
każda linia tekstu z pliku wejściowego
(o ile nie jest za długa)
będzie znajdować się w osobnym wierszu.
W efekcie każda linia pliku wejściowego
jest traktowana jako osobny akapit.
\quad(Na przykład plik wejściowy tego fragmentu ma postać:)
\smallskip}
```

\obeyspaces

Działa w taki sposób, że TeX nie ignoruje kolejnych spacji. Plik wejściowy tego fragmentu ma postać:

{\obeyspaces Działła w taki sposób, że \TeX\ nie ignoruje kolejnych spacji}. Plik ...

odstępny 8–15

- *Poziome odstępny w tekście.* Komendy `\negthinspace`, `\thinspace`, `\` (`\` i spacja), `\quad` i `\qquad` określają odstępny poziome. Efekt działania tych komend przedstawiony jest we wstępie, a każdą omówiono w dodatku. Czasem potrzebny jest mały odstęp „na końcu” *pochylej* odmiany pisma lub *kursywy*, abyśmy mieli pewność, że tekst nie „wejdzie” na odstęp przed kolejnym znakiem. Jest to tzw. *korekta do pisma pochylego* `\/`, np. `{\it korekta do pisma pochylego\}`. Poziome odstępny różnych rozmiarów można uzyskać za pomocą komendy `\hskip x`, gdzie `x` musi być podane w jednostkach rozpoznawanych przez `TeX`-a. Odstępny poziome, w miejscu których nie chcemy łamać wiersza, uzyskujemy za pomocą komendy `\kern`. `\hfil` i `\hfill` można wykorzystać do wypełnienia części wiersza justunkiem.
- *Odstępny pionowe.* Ustawienie `\baselineskip` kontroluje odstępny między wierszami. Komendy `\smallskip`, `\medskip` i `\bigskip` tworzą odstępny pionowe różnego rozmiaru. Wszystkie te komendy zostały omówione oddzielnie. Pionowe odstępny różnych rozmiarów można uzyskać za pomocą komendy `\vskip x`, gdzie `x` musi być podane w jednostkach rozpoznawanych przez `TeX`-a. `\vfil` i `\vfill` wykorzystuje się do wypełnienia części strony justunkiem.
- *Zagadnienia ogólne.* Niektóre z komend dotyczących odstępów mogą nie działać na początku wiersza czy strony (pracują tylko między fragmentami tekstu). Jest to dobre z wielu względów: np. w składzie długiej pionowej listy, gdzie każdy wpis listy odsunięty jest od poprzedniego o odległość `\bigskip`, odstępny potrzebne są tylko pomiędzy kolejnymi wpisami na tej samej stronie, nie zaś u góry nowej strony. Wyjściem z sytuacji jest umieszczenie na początku wiersza czy strony polecenia `\hbox{}` lub równoważnego `\null`. Alternatywą jest użycie `\hglue` lub `\vglue`.
- *Patrz także:* puste odstępny, `\kern`, odstępny między wierszami, matematyka: odstępny, `\obeyspaces`, nakładanie, akapity i położenie tekstu na stronie.

odstępny między wierszami

Parametr ten kontrolowany jest za pomocą komendy `\baselineskip`. Jego wartość domyślna w Plain `TeX`-u wynosi `12pt`. Dodatkowo są dwa parametry kontrolujące odstępny między wierszami. `\lineskiplimit` określa, jak blisko siebie mogą znaleźć się dwa wiersze. Wiersze, które znajdują się w odległości mniejszej niż wartość `\lineskiplimit`, są odsuwane na odległość `\lineskip` (komendy te omówione zostały w oddzielnych opisach). Instrukcja, jak zmieniać odstępny między wierszami, znajduje się pod hasłem `\offinterlineskip`. Patrz również `\openup`.

\offinterlineskip 48–49, 78–79

Usuwa odstęp pomiędzy liniami podstawowymi kolejnych wierszy. Efekt jest silniejszy niż w przypadku komendy `\baselineskip Opt`. `TeX` posiada parametr `\lineskip`, który zabezpiecza wiersze przed zbytym ściśnięciem. W Plain `TeX`-u parametr ten ma wartość domyślną `1pt` (tzn., że wiersze, które znajdują się zbyt blisko siebie, będą odsuwane na odległość 1 punktu, nawet jeśli `\baselineskip` ma wartość `Opt`). Komenda `\offinterlineskip` niweluje jego działanie. Ma to zastosowanie w składzie tabel czy innych wyrównań, które rządzą się zasadami pionowego wyrównania kilku poziomych linii. Istnieje również komenda `\nointerlineskip`, która likwiduje odstęp pomiędzy wierszami w miejscu jej zastosowania (musi więc być umieszczona w odpowiednim miejscu).

ograniczniki 42–43, 54–55, 60–63

Symbole, takie jak nawiasy, klamry, itd., wykorzystywane są (zwłaszcza w matematyce) do oddzielania części wyrażeń w tekście. Patrz *matematyka: ograniczniki*.

\oldstyle

Komenda `{\oldstyle 0 1 2 3 4 5 6 7 8 9}` składa 0123456789.

\omit * 48–49

Wykorzystywana w wyrównaniach, pozwala, aby wpis ignorował specyfikację z preambuły. Patrz *tabele i przykład VI*.

\openup

Zwiększa pionowe odstępy, np. `\openup2pt` dodaje 2pt do wszystkich parametrów dotyczących odstępów między wierszami. W Plain T_EX-u istnieją też jednostki `\jot`, w których może być wyspecyfikowana wartość `\openup` (`\jot` odpowiada wartości 3pt). Układy równań eksponowanych w T_EX-u mają wbudowaną standardowo wartość `\openup=\jot`. Można zmieniać wartość `\jot` (patrz `\jot`), aby regulować odstępy pomiędzy wierszami w przypadku wyrażeń eksponowanych. A oto przykład:

$$\begin{array}{l} x = y \\ y \approx z \end{array}$$

```
$$\openup-\jot % ODSTĘP ZREDUKOWANY O JEDNOSTKĘ ,,JOT''
\displaylines{x=y\cr y\approx z\cr}$$
```

\or *

Wykorzystywana w połączeniu z `\ifcase`. Przykład zastosowania obu komend znajduje się pod hasłem `\month`.

\output *

Ostatnią szansą na zmianę wyglądu stron wynikowych jest podgląd pliku `.dvi`. Użytkownik może wówczas dołożyć powtarzający się materiał na każdej ze stron lub spowodować narysowanie ramki (*ruled box*) wokół strony. Standardowa procedura wyjściowa (*output routine*) w Plain T_EX-u wykonuje kilka rzeczy: umieszcza nagłówek na szczycie i stopkę na dole `\vbox`-a reprezentującego bieżącą stronę; pomiędzy umieszcza `\vbox` reprezentujący stronę. Powiększa także numer strony i „wypuszcza stronę”. Przykłady programów wyjściowych znajdują się w uwagach końcowych.

\$\$\over\$ * 28–29, 38–43, 52–55, 60–63

Składa ułamki. Patrz *przykład II*.

\$\$\overbrace\$ 24–25, 62–63

Umieszcza nawias nad wyrażeniem (patrz *przykłady II i IX, równanie 9*).

\$\$\overleftarrow\$

Można umieścić strzałkę nad tekstem. A oto plik wejściowy:

```
$$\overleftarrow{\hbox{Można umieścić strzałkę nad tekstem.}}$
```

`\overline` \$ 30–31

Składa kreskę nad wyrażeniem (nadmaksowanie). Na przykład konstrukcja $\overline{a^2 + b^2 = c^2}$ powstała w następujący sposób:

Na przykład konstrukcja `\overline{a^2+b^2=c^2}` powstała...

`\overrightarrow` \$

Działa tak jak `\overleftarrow`.

`\pagegoal` *

Kiedy \TeX składa materiał pionowo, stara się trzymać pionowego rozmiaru umieszczonego w liczniku `\pagetotal`. Utrzymuje rozmiar `\pagetotal`, jeśli to jest możliwe. `\pagegoal` na początku ma wartość `\vsize`, ale można zmieniać tę wartość w celu uzyskania zabawnych efektów. W tym miejscu wartość `\pagegoal` wynosi 621.52243pt, i wartość `\vsize` wynosi 621.52243pt. Wartości te uzyskano za pomocą komend `\the\pagegoal` i `\the\vsize`.

`\pageinsert` 30–33

Wstawia stronę z materiałem (np. wykresem, tabelą czy pustą stroną) w miejsce następnej wolnej strony (patrz przykład III).

`\pageno` vi–xiii, 2, 80–81

Nazwa rejestru (licznika), w którym \TeX umieszcza numer bieżącej strony. Ten rejestr to w tej chwili `\count0` (patrz `\count`), ale przyjęto odnosić się do niego przez nadaną nazwę `\pageno`. Patrz str. 60 (ustawienie numeru stron) lub str. vi i viii (numeracja rzymska stron).

`\pagetotal` *

Określa całkowity pionowy rozmiar materiału zgromadzonego w danym momencie na danej stronie. Wartość ta w tym miejscu wynosi 356.43333pt. (Ponieważ \TeX czyta materiał, zanim złoży stronę, jest to w tym przypadku rozmiar tekstu z poprzedniej strony). Kiedy parametr ten przekroczy wartość `\pagegoal`, \TeX wie, że należy rozpocząć nową stronę.

Wartość `\pagetotal` zmieniła się po złożeniu ostatniego akapitu: wynosi teraz 417.94444pt (uzyskano ją za pomocą komendy `\the\pagetotal`).

`\par` * viii–ix, 12–13, 18–19, 58–59, 68–71, 74–75, 84–85

Kończy akapit. Komendę tę \TeX dokłada na końcu każdego czytanego akapitu (zanim przystąpi do dalszego przetwarzania dokumentu), nawet jeśli stosowany jest inny mechanizm zakańczania akapitu (jak np. pusta linia w pliku wejściowym). Pozwala to na przedefiniowanie komendy `\par` w celu umieszczenia dowolnego znaku czy symbolu na końcu każdego akapitu. Patrz `\endgraf` oraz przykład XIII.

`\parfillskip` * 18–19

\TeX czyta cały akapit, zanim go złoży. Pozwala to kontrolować nie tylko początek (np. ustawienie wcięcia), ale i koniec akapitu za pomocą komendy `\parfillskip`. Wartość domyślna tego parametru wynosi `Opt plus 1fil`; tzn. na końcu akapitu nie jest dokładany ekstra odstęp, natomiast umieszczany jest klej (*glue*) do wypełnienia ostatniego wiersza justunkiem. Parametr `\parfillskip` został zmieniony na początku tego akapitu, tak aby zakończyć akapit 1 cal od prawego marginesu. A oto plik wejściowy tego akapitu:

`\parindent`

```
{\parfillskip 1in \TeX\ czyta cały akapit
...
A oto plik wejściowy tego akapitu:\par}
```

`\parindent` * [viii-ix](#), [6-7](#), [22-23](#), [66-71](#)

Ustawia wielkość wcięcia akapitu. Wartością domyślną w Plain \TeX -u jest 20pt. Ujemna wartość powoduje, że pierwszy wiersz jest wysunięty, nie jest wcinany. Więcej informacji na ten temat można znaleźć pod hasłem `akapity`.

`\parshape` * [22-23](#)

Dopasowuje kształt akapitu (patrz ostatnie akapity w przykładzie I).

`\parskip` * [viii-ix](#), [xiv-1](#), [68-71](#), [80](#)

Ustawia odległość między akapitami. W Plain \TeX -u jest to `\Opt plus 1pt`, gdzie `plus` pozwala na rozciągnięcie tego odstępu, jeśli to konieczne. Wartość rozciągnięcia może być większa, można także określić kurczliwość odstępu, za pomocą komendy `minus`. Patrz `\bigskip`, `\medskip` i `\smallskip` — inne przykłady elastycznych odstępów pionowych.

`pauzy` [vix-3](#)

Aby otrzymać:	-	—	—	—
należy wpisać:	-	--	---	\$\$-

Są to, odpowiednio: dywiz, półpauza (*en-dash*), pauza/myślnik (*em-dash*) i minus matematyczny.

`\phantom` [40-41](#), [62-63](#)

`` liczy wysokość (`height`), głębokość (`depth`) i szerokość (`width`) *tekstu*, ale go nie drukuje; wstawia w jego miejsce odstęp danego rozmiaru. Patrz `\hphantom` i `\vphantom`.

`pierwiastki` [24-25](#), [28-29](#), [62-63](#)

Symbole pierwiastka mogą być złożone instrukcjami `\root` oraz `\sqrt` (patrz przykłady II i IX).

`pismo grube`

Patrz `\bf`.

Plain \TeX

Pakiet komend opartych na podstawowych komendach \TeX -a. Pakiet ten jest powszechnie dostępny, tak jak i \TeX ; ludzie zwykle, kiedy mówią \TeX , mają na myśli Plain \TeX -a. Dowolna komenda wymieniona w dodatku, nie oznaczona gwiazdką *, jest komendą Plain \TeX -a i jest skrótem zestawu komend podstawowych. Dodatek B książki Donalda Knutha *TeXbook* zawiera definicje (za pomocą pojęć podstawowych) wszystkich komend Plain \TeX -a.

Domyślny format \TeX -a jest w rzeczywistości formatem Plain \TeX -a. Jest on idealny do składu wydruków próbnych i raportów technicznych. Do innych typów dokumentów format musi być przerobiony lub wykorzystany całkiem inny pakiet.

plus xiv-1, 11, 50

Określa rozciągliwość kleju (*glue*); np. komenda `\hskip 10pt plus 5pt` wstawia odstęp o wielkości 10–15 punktów. (Klej może być rozciągnięty poza wyspecyfikowane limity — ale nie ściśnięty poza te granice.) Patrz *minus* i *słowa kluczowe*.

pochodne

Patrz przykłady V i VII.

podkreślenia

W tekście — patrz `\underbar`; we wzorach matematycznych — `$_underline$` oraz *matematyka: akcenty*.

podział wierszy x-xi, 2, 8-9, 22, 26, 34

Podział wiersza można spowodować komendą `\hfil\break` (lub `\allowbreak`). Wyrażenie postaci „tak/nie” nie zostanie złamane w miejscu znaku /, jeśli wpisujemy `tak/nie`, a dopiero w przypadku wpisu `tak/slash nie`. Łamanie wiersza jest dozwolone w miejscu występowania odstępów `\` (`\` i pojedynczy odstęp), `\quad`, `\qqquad` lub `\hskip`. Do wskazania możliwego miejsca złamania wiersza można wykorzystać łącznik dyskrecjonarny (patrz *dzielenie wyrazów*).

Aby zapobiec łamaniu wierszy, wykorzystuje się kilka technik: (1) Pudełko poziome `\hbox`. Materiał umieszczony w pudełku poziomym nie jest dzielony; np. komendy odtwarzające *verbatim* (dosłowny skład) tego dodatku (lub uwagi pod ramkami z plikami wejściowymi) umieszczone są w `\hbox`-ie. (2) *Wiązania* (*ties*). Słowa mogą być połączone razem za pomocą komendy znakowej `~`. Składa ona pojedynczy odstęp w miejscu, w którym podział wiersza jest zabroniony. Wiązania są przydatne w kilku przypadkach: gdy odnosimy się do nazwy (np. twierdzenie 3, lub rozdział B), gdy odnosimy się do zmiennej w wyrażeniu (np. wysokość *h*), gdy wymieniamy przypadki w akapicie (patrz część pliku wejściowego tego akapitu, poniżej) [a w języku polskim także do likwidowania spójników na końcu wiersza — przyp. tłum.]. Generalnie łączniki powinny być stosowane zawsze w miejscach, w których złamanie wiersza jest niewskazane. (3) Komenda `\nobreak`. Komenda wykorzystywana w poziomym trybie pracy (w wierszu a nie między wierszami); podział wiersza jest zabroniony w miejscu za tą komendą. (4) Komenda `\kern`. Wiersz nie będzie dzielony w miejscu odstępu umieszczonego przez tę komendę lub inne komendy na niej oparte: `\enspace`, `\thinspace`, `\negthinspace`.

Oto część pliku wejściowego powyższego akapitu:

```
... umieszczone są w "\hbox"-ie. (2)~{\s1 Wiązania\} ({\it ties\}).
Słowa mogą być połączone...
```

Patrz również *matematyka: podział wierszy*.

położenie stron

Patrz `\hoffset` oraz `\voffset`.

położenie tekstu na stronie

Patrz przykłady I, VIII, XV. Komendy `\raise` i `\lower`, przedstawione w pliku wejściowym równania 5 przykładu IX i przykładu XV, mogą być wykorzystane do podniesienia czy obniżenia pudełka z tekstem (patrz *pudełka*). Są też podobne komendy służące do przemieszczania pudełek poziomo: `\moveleft` i `\moveright`. Patrz także poszczególne wpisy w dodatku (jak np. *wyśrodkowanie* czy *akapity*).

powiększanie [viii–ix](#), [xiv–1](#), [64–65](#), [80–81](#)

Patrz `\magnification` (powiększanie całości dokumentu) oraz fonty i `\magstep` (kroki powiększenia wewnątrz dokumentu). Rozmiar bez powiększenia reprezentuje liczba 1000, stąd font skalowany na 500 jest o połowę mniejszy, a skalowany na 3000 — trzy razy większy. Instrukcją `\font\huge=plr10 scaled 5000` można uzyskać 10-punktową antykwę Computer Modern PL, powiększoną pięć razy (o nazwie `\huge`).

Urządzenia drukujące nie zawsze przyjmują wszystkie stopnie pisma; zestaw rozmiarów uzyskuje się przez pomnożenie liczby 1000 przez odpowiednią potęgę liczby 1,2; np. 1200, 1440 itd. \TeX określa powiększenie maksymalnie do czterech cyfr (patrz dyskusja pod hasłem `\magstep`). Problemy pojawiają się, jeśli stopień pisma zwiększany jest wewnątrz powiększanego dokumentu. Na przykład dokument o `\magstep4` jest skalowany na 2074 (ale „właściwa” wartość wynosi $1000 \times 1,2^4 = 2073,6$). Wewnątrz takiego dokumentu powiększenie fontu do `\magstep1` może spowodować, że urządzenie drukujące będzie skalować na wielkość 2489 ($2074 \times 1,2 = 2488,8$), podczas gdy maksymalne dostępne skalowanie to 2488. W takich przypadkach, aby określić właściwy rozmiar, należy z komendą `\font` wykorzystać słowo kluczowe `at`. Patrz przykład X oraz definicja komendy `\title` w pliku wejściowym do str. [viii](#).

powiększenia

Patrz fonty oraz powiększanie.

`\pretolerance` * [86–87](#)

Komenda określająca tolerancję pustego odstępu, jaki \TeX wstawia przy podziale słów. (\TeX najpierw usiłuje złożyć akapit bez dzielenia słów i tylko jeśli nie znajdzie właściwej struktury, dzieli słowa.) Wartość ta może być z przedziału pomiędzy -1 (natychmiast dzieli słowa) a 1000 (nie pozwala na podział słów, chyba że są słowa dłuższe niż wiersz). Wartość domyślna w Plain \TeX -u wynosi 100. Patrz plik wejściowy podziękowań.

`\prevgraf` *

Kontroluje numerację wierszy we właśnie składanym akapicie. Parametr ten jest automatycznie ustawiany na 0 na początku każdego akapitu. W przypadku komendy składającej dwukolumnowe uwagi na stronach wejściowych głównej części tej książki, `\prevgraf` wykorzystana została do kontroli całkowitej liczby wierszy: jeśli liczba ta jest parzysta, wiersze są dzielone na dwie kolumny w jeden sposób, jeśli nieparzysta — w inny (patrz uwagi końcowe). Aby uzyskać żądany wygląd strony czy dokumentu, można zmieniać wartość `\looseness`.

Przed tym akapitem `\prevgraf` ma wartość 7. Plik wejściowy tego fragmentu ma postać:

```
{\count255=\prevgraf \ind Przed tym akapitem
"\prevgraf" ma wartość \the\count255\}. Plik ...
%UWAGA: \ind jest komendą zdefiniowaną przez autora.
```

przepełnione pudełka

Patrz błędy i pudełka.

pt [viii–ix](#), [xiv–1](#), [9](#), [11–13](#)

Patrz punkt i słowa kluczowe.

pudełka 74–75

\TeX organizuje materiał w kwadratowe struktury zwane pudełkami (*boxes*). Są dwa typy pudełek: poziome (`\hbox`) i pionowe (`\vbox`), które organizują materiał w różny sposób i inaczej dopasowują się do innych pudełek (patrz przykład XV).

\TeX tak samo traktuje pojedyncze znaki, jak i kompleksowe struktury tekstu i wzorów, dzieli je na prostokąty o określonym rozmiarze. Aby zrozumieć, jak program działa, należy zapoznać się ze sposobem składu takich pudełek. Szczegółowe wyjaśnienie tego problemu można znaleźć w książce Knutha *TEXbook*.

Odległość między pudełkami \TeX wypełnia klejem (*glue*). Konstrukcja ta jest omówiona osobno. Klej ten jest elastyczny, pozwala odstępom między pudełkami kurczyć się i rozciągać, wewnątrz wcześniej wyspecyfikowanych granic. Prostokątne pudełka i elastyczny klej tworzą model, na którym opiera się \TeX .

Przykład XV ilustruje zastosowanie pudełek. Jedna z podstawowych cech pudełek (materiał wewnątrz pudełka nie jest dzielony) została wykorzystana w kilku miejscach tej książki. Wewnątrz `\hbox`-a zabroniony jest podział wiersza, wewnątrz `\vbox`-a nie dopuszcza się łamania kolumn. Na przykład tabele 1–4 w przykładzie VI zostały umieszczone w pudełkach pionowych, aby zapobiec ich podziałowi na strony.

Poziomy rozmiar pudełka nazywany jest jego szerokością (*width*). Punkt w lewym dolnym brzegu pudełka jest jego punktem odniesienia. Linia pozioma, przechodząca przez ten punkt, jest linią podstawową (*baseline*) pudełka. Linie podstawowe pudełek zawierających pojedyncze znaki wykorzystywane są do wyrównywania tych znaków; linie podstawowe większych pudełek są określone według linii podstawowych ich zawartości, z wykorzystaniem zestawu precyzyjnych reguł. Patrz `\vbox` i `\vtop`. Pionowe rozmiary pudełka nad i pod jego linią podstawową to, odpowiednio, jego wysokość (*height*) i głębokość (*depth*).

Pudełka układane poziomo są wyrównywane według ich linii podstawowych. Przy składzie pionowym, bezpośrednio jeden nad drugim umieszczane są punkty odniesienia pudełek. Pudełko jest traktowane jako niepełne (*underfull*), jeśli jest większe niż całkowity rozmiar jego zawartości, a jako przepełnione (*overfull*), jeśli zbyt wiele materiału w nim umieszczono. Zawartość pudełka może być albo innym pudełkiem, albo justunkiem (np. klejem).

punkt

Jednostka powszechnie wykorzystywana do specyfikowania wielkości, takich jak wcięcie akapitu czy odstęp między wierszami. Jeden cal to 72,27 punktu. Punkt jest wykorzystywany jako słowo kluczowe `pt`.

puste odstępy 10–13, 32–33

- Pojedynczy znak odstępu można uzyskać za pomocą instrukcji `\` (`\` i *spacja*). Informacji na temat odstępów innej szerokości należy szukać w opisach `\obeyspaces` i `\hskip`.

- Wpis `\hfil\break\line{}` składa pusty wiersz. `\hfil\break` kończy wiersz bieżący, a `\line{}` oznacza pustą linię. Aby uzyskać pusty wiersz w składzie w przypadku, gdy \TeX znajduje się na początku wiersza, ale wewnątrz akapitu (nie rozpoczyna się akapitu), można wykorzystać komendę `\line{}`. Aby uzyskać pusty wiersz, bez wyraźnego zakończenia bieżącego wiersza, można w dowolnym miejscu

umieścić komendę `\vadjust{\hfil\break\line{}}`. By pozostawić pustą pierwszą linię na początku nowego akapitu, należy zastosować komendę `\hfil\break` lub `\line{}`. Pierwsza z komend „pochłonie” wcięcie akapitu (następny wiersz nie

`\qqad`

będzie wcięty), druga przeniesie je do następnego wiersza.

Plik wejściowy powyższego fragmentu ma postać:

```
... pusty wiersz,\vadjust{\hfil\break\line{}} bez wyraźnego ...
```

- W wyniku działania instrukcji `\vfil\eject\line{}\vfil\eject` uzyskuje się w składzie pustą stronę (komenda ta najpierw kończy stronę bieżącą). Na początku strony należy wykorzystać komendę `\line{}\vfil\eject`. Aby uzyskać pustą stronę, bez nagłego zakończenia strony bieżącej, można zastosować komendę `\pageinsert` z wstawką w postaci `\vfil`.
- Więcej informacji na temat odstępów można znaleźć pod hasłem `\null` i odstęp.

`\qqad` 10–11, 14–15

Wstawia odstęp takiej[] wielkości (`\odstęp takiej[\qqad]wielkości`). Komenda ta jest zdefiniowana w Plain \TeX -u następująco:

```
\def\qqad{\hskip2em\relax}
```

Patrz `\quad`.

`\quad` 10–11, 14–15

Wstawia odstęp takiej[] wielkości (`\odstęp takiej[\quad]wielkości`). Komenda ta jest zdefiniowana w Plain \TeX -u następująco:

```
\def\quad{\hskip1em\relax}
```

Komendy `em` i `\relax` są zdefiniowane osobno.

`\raggedbottom` 86–87

Plain \TeX normalnie wyrównuje linie podstawowe kolejnych stron. Oznacza to, że pionowe odstępy wewnątrz strony mogą być trochę elastyczne. Ta elastyczność jest wbudowana w komendę `\parskip`, jak również w inne komendy dotyczące odstępów pionowych, np. `\bigskip`. `\raggedbottom` odwraca sytuację, powodując, że pionowe odstępy na stronie są nieco sztywniejsze, a spód strony ruchomy (*ragged bottoms*). Instrukcja ta może być wyłączona za pomocą komendy `\normalbottom`. Patrz także `\raggedright`.

`\raggedright` 86–87

Skład „w chorągiewkę” — wyłącza normalne prawostronne wyrównanie tekstu przez umieszczenie sztywnych odstępów między wyrazami (normalnie są one elastyczne). Patrz podziękowania.

`\raise` * 60–61, 74–75

Podnosi pudełka. Komendy `\raise` i `\lower` mogą być wykorzystywane, jeśli \TeX pracuje w trybie poziomym. Patrz tytuł przykładu XV.

rejstry 68

\TeX może umieszczać różne typy informacji w rejestrach. Są specyficzne klasy rejestrów, dostępne do umieszczania specyficznych typów informacji. Ta książka przedstawia trzy z nich, są jednak jeszcze inne. Rejestry wykorzystywane do umieszczania liczb całkowitych to rejestry `\count`, te, w których przechowuje się wymiary, to rejestry `\dimen`, a te, w których umieszcza się tekst, to rejestry `\box`. Każdy z typów rejestrów został omówiony osobno.

`\relax` *

Komenda oznaczająca „nie rób nic”. Jedną z sytuacji jej zastosowania jest połączenie komend ze słowami, które T_EX może potraktować jako kontynuację komendy. Na przykład jeśli słowo `height` zostanie umieszczone zaraz za komendą `\hrule`, należy wpisać `\hrule\relax`, inaczej T_EX zinterpretuje `height` jako słowo kluczowe, oznaczenie wysokości `\hrule`, i będzie szukał cyfry. Jeśli jej nie znajdzie, wyświetli komunikat *missing number*. Podobnie w przypadku komend `skip`, problem mogą stwarzać słowa `plus` i `minus`. Patrz błędy i słowa kluczowe.

`\repeat` 68–69, 72–73, 78–79

Wykorzystana w połączeniu z `\loop` do wykonywania powtarzalnych zadań.

`$$\right$` * 40–43, 60–63

Tworzy prawy ogranicznik odpowiedniego rozmiaru w eksponowanym wyrażeniu matematycznym. Może być wykorzystany wyłącznie w połączeniu z `$$\left$`; pod tym wpisem znajduje się dalsze omówienie.

`\rightline` viii–1, 66–67, 84–87

Dosuwa tekst w wierszu do prawego marginesu.

Plik wejściowy tego wiersza ma postać:

```
\rightline{Dosuwa tekst w wierszu do prawego marginesu.}
```

`\rightskip` * 20–23, 70–71

Przemieszcza prawy margines tekstu do wewnątrz lub na zewnątrz (patrz przykład I). Wartość `\rightskip` wykorzystana w składzie akapitu jest jedyną wielkością narzuconą w punkcie, w którym akapit się kończy. Patrz akapity.

`\rlap` 2, 36–37

Nakłada materiał w prawą stronę (np. w celu umieszczenia go na prawym marginesie). Na przykład znak Y powstał dzięki komendzie `\rlap{---}Y` (patrz przykład IV oraz definicja komendy `\headline` z dodatku, zamieszczona w uwagach końcowych).

`\rm` 8–9, 22–25, 40–41, 44–45, 52–55, 62–65, 70–71, 86–87

Składa tekst antykwą (domyślny krój pisma w Plain T_EX-u).

`\romannumeral` *

Dokonuje konwersji cyfr arabskich na rzymskie. Na przykład numer bieżącej strony to CXLV w tej numeracji. Plik wejściowy tego fragmentu ma postać:

```
to \uppercase\expandafter{\romannumeral\pageno} w tej numeracji.  
Plik...
```

`\expandafter` to sztuczka, powodująca, że `\uppercase` działa na rozwiniętą wcześniej komendę `\romannumeral` (tzn. po konwersji numeru strony na numerację rzymską). Zestaw `\romannumeral\pageno` składa cyfry rzymskie ale małymi literami.

rozmiar

Patrz fonty, powiększanie oraz jednostki.

rozmiar strony

Patrz `\hsize` i `\vsize`.

rozszerzanie (spread)

Patrz słowa kluczowe, `\hbox` i `\vbox`.

scaled [xiv-1](#), [64-65](#), [80-81](#)

Patrz słowa kluczowe.

`\scriptscriptstyle` * [14-15](#), [62](#)

Domyślny styl (krój pisma o stopniu zwykle 5 punktów) indeksów do indeksów.

`\scriptstyle` * [14-15](#), [40-41](#), [60-61](#)

Domyślny styl (krój pisma o stopniu zwykle 7 punktów) indeksów.

`\setbox` * [80](#)

Umieszcza materiał w rejestrach `\box`. Przykłady można znaleźć pod hasłami `\box` oraz `\newbox`.

`\settabs` [42](#), [44-45](#), [50-51](#)

Tworzy proste wyrównania. Patrz tabele I i IV w przykładzie VI oraz [tabulowanie](#).

`\shipout` *

Komenda, która przesyła stronę do pliku `dvi`. Patrz określenie `\output` w uwagach końcowych.

`\skew$`

Wyrównuje podwójne akcenty. Patrz [matematyka: odstępy](#).

`\sl` [viii-ix](#), [xiv-1](#), [8-9](#), [20-21](#), [86-87](#)

Składa *pochyły krój pisma* (`{\sl pochyły krój pisma}`).

słowa kluczowe

Słowa, które działają jako kontynuacja (czy modyfikacja) odpowiednich komend, kiedy wykorzystywane są w połączeniu z nimi. Oto kompletna lista słów kluczowych:

- `bp`, `cc`, `cm`, `dd`, `in`, `mm`, `pc`, `pt`, `sp`; patrz jednostki.
- `em`, `ex`; patrz oddzielne opisy każdego z nich oraz jednostki.
- `at`, `scaled`: wykorzystywane w połączeniu z komendą `\font` do określania rozmiaru fontu; patrz przykład X i fonty.
- `by`: wykorzystywane z `\multiply`, `\divide` lub `\advance`, do mnożenia, dzielenia lub zwiększania wartości rejestru (licznika).
- `depth`, `height`, `width`: określają rozmiar kresek; patrz [kreska \(rule\)](#).
- `fil`: specjalna jednostka wykorzystywana do wyspecyfikowania elastycznego kleju (*glue*). Na przykład `\hfil` oznacza to samo co `\hskip0pt plus1fil`, `\hss` to samo co `\hskip0pt plus1fil minus1fil`, a `\hfilneg` jest odpowiednikiem `\hskip0pt plus-1fil`. Patrz [\parfillskip](#).
- `l`: połączone z `fil` (w celu utworzenia `fill` lub `filll`) tworzy silniejsze komendy; np. komenda `\hfill` jest równoważna `\hskip0pt plus1fill`.

- **minus, plus:** określają elastyczność odstępów; patrz klej.
- **mu:** matematyczna jednostka kleju. Konieczna jest tu specyficzna jednostka, ponieważ czasem w tym samym wzorze wykorzystywane są różne stopnie krojów pisma. Jednostka dopasowuje się automatycznie do rozmiaru **em**; $18\mu = 1em$. Jest wykorzystywana do określania rozmiaru odstępów poziomych we wzorach, za pomocą komendy **\mskip**. Na przykład wąski ujemny odstęp w trybie matematycznym (**\!**) jest zdefiniowany w Plain T_EX-u jako **\mskip -3mu**. Inne komendy dotyczące odstępów: **\,**, **\>** i **\;**; mają rozmiary, odpowiednio: **3mu**, **4mu plus2mu minus4mu** oraz **5mu plus5mu**.
- **spread, to:** określają rozmiary pudełek; patrz **\hbox**, **\vbox** i przykład XV.
- **true:** określa rozmiary niezależne od powiększenia; patrz strona *xiv* i jednostki.

\smallbreak 34–37

Sugeruje T_EX-owi złamanie kolumny lub umieszczenie odstępu pionowego o wielkości **\smallskip**.

\smallskip 6–7, 11–13, 20–21, 48–49, 54–55

Składa odstęp $\rule{0.5cm}{0.4pt}$. **\smallskipamount**, standardowy rozmiar tego odstępu, jest zdefiniowany jako **3pt plus 1pt minus1pt**, czyli 3-punktowy odstęp, który może skurczyć się lub rozciągnąć, jeśli to konieczne. **\smallskip** jest zdefiniowana jako **\vskip\smallskipamount**. Patrz **\bigskip**.

\smash 62–63

Komenda **\smash{tekst}** drukuje tekst po przypisaniu mu zerowej wysokości i głębokości. **\smash** jest przeciwieństwem **\vphantom** i często jest wykorzystywana w połączeniu z tą komendą. **\smash** wykorzystana w tym miejscu składa podkreślenie umieszczone za wysoko (ponieważ T_EX przyjmuje, że wyrażenie ma zerową wysokość): $\int_0^1 dx = 1$. Plik wejściowy tego fragmentu ma postać:

```
wysokość): $\overline{\smash{\int_0^1 dx=1}}$.
```

\span *

Spełnia dwa zadania przy tworzeniu tabel. Jeśli występuje w preambule, powoduje, że komendy pojawiające się dalej ulegają rozwinięciu (czyli zastępowane są własnymi definicjami). Jeśli pojawia się we wpisie, łączy kolejne kolumny. Komenda **\multispan** ma wbudowaną komendę **\span**. Patrz tabele.

\special *

Komenda nie interpretowana przez T_EX-a, a umieszczana w pliku **dvi** wraz z argumentem zrozumiałym dla drukarki. Instruuje drukarkę do wykonania specjalnych czynności, jak np. importu grafiki, wstawki języka PostScript, itp. Patrz wykresy.

\string * 6–7, 10–13, 16–19, 68–69

Określa komendę wskazującą, że następująca po niej sekwencja znaków jest jej nazwą. Jest to łatwy sposób drukowania komend. Na przykład **\tt\string\end** składa **\end**, a **\tt\string{}** znak **{**. **\tt** jest tu ważne, bez tej komendy *backslash* na początku komendy zostałby wymieniony na inny znak (ponieważ kody znakowe odpowiadają różnym znakom w różnych krojach pisma).

strony

Pod hasłami `\pagegoal` i `\pagetotal` znajduje się krótki opis, jak \TeX składa stronę (są tam różne szczegóły nie omawiane tutaj). Inne aspekty składu stron, takie jak łamanie stron, rozmiar strony, położenie strony itd., są omówione pod osobnymi wpisami.

`\strut` 48–49

Niewidzialny `\box` o wysokości 8,5 pt i głębokości 3,5 pt. Wykorzystywany do rozdzielania wierszy, które są zbyt blisko siebie, lub do upewnienia się, że odstępy między liniami są regularne (np. w tabeli; patrz tabela 3 w przykładzie VI).

szerokość (width) vi–vii, 12–13, 76–85

Odnosi się do szerokości pudełek i kresek tworzonych przez \TeX -a. Patrz `\wd`. Jest to także słowo kluczowe, wykorzystywane do wyspecyfikowania szerokości kresek.

tabele 42–51

W przykładzie VI pokazano, jak składać tabele. Przedstawione tam metody są pojęte, ale niewygodne do rutynowych zadań. Poniższe wiersze zawierają definicje makr ułatwiających konstrukcję tabel. Użytkownikowi te definicje mogą się wydawać trudne, zwłaszcza jeśli rozpoczyna pracę z \TeX -em. Jednak wszystko, co trzeba zrobić, to skopiować te linie i wykorzystać definicje w prosty, pokazany dalej sposób:

```
%***** DEFINICJE KOMEND DLA TABEL *****
\newdimen\tempdim
\newdimen\othick \othick=,4pt
\newdimen\ithick \ithick=,4pt
\newdimen\spacing \spacing=9pt
\newdimen\abovehr \abovehr=6pt
\newdimen\belowhr \belowhr=8pt
\newdimen\nexttovr \nexttovr=8pt

\def\r{\hfil&\omit\vrsp\vrule width\othick\cr&}
\def\rr{\hfil\down{\abovehr}&\omit\vrsp\vrule width\othick\cr
\noalign{\hrule height\ithick}\up{\belowhr}&}
\def\up#1{\tempdim=#1\advance\tempdim by1ex
\vrule height\tempdim width0pt depth0pt}
\def\down#1{\vrule height0pt depth#1 width0pt}
\def\large#1#2{\setbox0=\vtop{\hsize#1 \lineskiplimit=0pt \lineskip=1pt
\baselineskip\spacing \advance\baselineskip by 3pt \noindent
#2}\tempdim=\dp0\advance\tempdim by \abovehr\box0\down{\tempdim}}
\def\dig{\hphantom0}
\def\hgap#1{\hskip-\nexttovr\hskip#1\hskip-\nexttovr\relax}
\def\vrsp{\hskip\nexttovr\relax}
\def\toprule#1{\def\startrule{\hrule height#1\relax}}
\toprule{\othick}
\def\nstrut{\vrule height\spacing depth3,5pt width0pt}
\def\exclaim{\char'!}
\def\preamble#1{\def\startup{#1}}
\preamble{&##}
{\catcode'\!=\active
\gdef!\{\hfil\vrule width0pt\vrsp\vrule width\ithick\relax\vrsp&}}

\def\table #1{\vbox\bgroup \setbox0=\hbox{#1}
\vbox\bgroup\offinterlineskip \catcode'\!=\active
```

```

\halign\bgroup##\vrule width\othick\vrsp&\span\startup\ostrut\cr
\noalign{\medskip}
\noalign{\startrule}\up{\belowhr}&}

\def\caption #1{\down{\abovehr}&\omit\vrsp\vrule width\othick\cr
\noalign{\hrule height\othick}\egroup\egroup \setbox1=\lastbox
\tempdim=\wd1 \hbox to\tempdim{\hfil \box0 \hfil} \box1 \smallskip
\hbox to\tempdim{\advance\tempdim by-20pt\hfil\ vbox{\hsize\tempdim
\noindent #1}\hfil}\egroup}
%*****KONIEC DEFINICJI*****

```

Najgorsza część za nami.

A

1	2	3
Jeden	Dwa	Trzy
Cztery	Pięć	Sześć

Łatwe, prawda?

Ta tabela powstała z pliku wejściowego:

```

$$\table{\bf A}
1      ! 2      ! 3      \rr
Jeden ! Dwa   ! Trzy \r
Cztery! Pięć ! Sześć
\caption{\sl Łatwe, prawda?}$$

```

Wpisy w tabeli są umieszczone pomiędzy `\table{}` i `\caption{}`, wpisy w tym samym wierszu są oddzielane znakiem `!`, a koniec wiersza oznacza się `\rr` (jeśli chce się uzyskać pionową kreskę) lub `\r`; dotyczy to wszystkich wierszy z wyjątkiem ostatniego. (Znaki `$$` użyte są do wyrównania tabeli i wstawienia odstępów nad i pod tabelą; są opcjonalne.) Można także zbudować bardziej złożoną tabelę:

B

1	2	3
Oto <i>test</i> . <i>Jak to wygląda?</i>	α	
Akceptowalne?		Tak/Nie

Możesz dodać tak długi podpis jak chcesz!

Plik wejściowy:

```

\medskip\hfil
\table{\bf B}
\hfil\bf 1      !\hfil\bf 2!\hfil\bf 3\rr
\hfill Oto {\sl test}!$\alpha$ !      \r
\it Jak to wygląda? !      !      \rr
\multispan{2} Akceptowalne? hfil !Tak/Nie
\caption{Możesz dodać tak długi podpis jak chcesz\exclaim}
\medskip

```

Można wykorzystać komendę `\multispan` wewnątrz tabel i tak jak wcześniej wyśrodkować wpisy, dosunąć je do prawej, zmienić krój pisma. `\exclaim` na końcu podpisu składa znak `!`; jest tu potrzebna specjalna komenda, gdyż znak wykrzyknika był zarezerwowany do innych celów — jako separator. Aby wyśrodkować tabelę, można otoczyć ją podwójnymi znakami `$$` (jw.) lub postąpić tak, jak w metodzie opisanej powyżej (wpisać np. `\medskip\hfil` przed tabelą i `\medskip` za nią). Poniżej przedstawiono przykłady tworzenia własnej preambuły:

C

Kol. 1	Kol. 2
<i>Jeden</i>	12344,01
<i>Dwa</i>	344,1
<i>Trzy</i>	126,9
<i>Cztery</i>	<i>Właśnie aby pokazać Ci, jak tworzyć wpisy wielo- wierszowe</i>

Plik wejściowy:

```

$$\preamble{\hfill\it ##\hfill\strut&###\hfill\sl ##&###}
\table{\bf C}
\bf Kol. 1&!\bf Kol. 2\hfill \rr
Jeden &! 12344,01\r
Dwa &! 344,1\dig\rr
Trzy && 126,9\dig\rr
Cztery &!\large{1,5in}{Właśnie aby pokazać Ci, jak tworzyć wpisy
wielowierszowe}
\caption{}$$

```

Określanie preambuły może wydać się trochę oszukane. W preambule muszą być ekstramiejsca na wszystkie pionowe linie; należy uważać przy grupowaniu materiału, aby zapobiec pomieszaniu się komend. W celu ułatwienia pracy wprowadzono w tabeli powyżej dwie komendy: `\dig` wstawia odstęp poziomy o szerokości cyfry, a `\large{}{}` umieszcza w tabeli materiał o większym rozmiarze. `\dig` może być wykorzystywana do wyrównywania kolumn cyfr względem przecinka dziesiętnego. Aby zastosować komendę `\large`, trzeba określić szerokość (około `1,6in`).

Można składać także inne wyrównania. Grubość kresek określają komendy `\othick` (grubość kresek zewnętrznych, domyślnie 0,4 punktu) i `\ithick` (grubość kresek wewnętrznych, domyślnie 0,4 punktu). Minimalna wysokość każdej z kresek jest określona za pomocą komendy `\spacing` (standardowo 9 punktów); odstęp nad i pod kreską poziomą ustawiają komendy `\abovehr` (domyślnie 6 punktów) i `\belowhr` (domyślnie 8 punktów); odstęp między kreskami pionowymi określa `\nexttobr` (domyślnie 8 punktów). Wszystkie te parametry mogą być zmienione, jak np. poniżej:

1	2		1	2
3	4	E	3	4
D			F	

Plik wejściowy tego fragmentu ma postać:

```
\line{\othickness=2pt
\table{ 1!2\rr 3!4\caption{\centerline{\bf D}} }
\hfil
{\ithick=,8pt \othickness=0pt \spacing=12pt \abovehr=18pt \belowhr=20pt
\nexttovr=30pt
\table{\bf E} 1!2\rr 3!4\caption{} }
\hfil
{\toprule{2pt}
\table{ 1 !\hgap{1pt}! 2\rr 3 !\hgap{1pt}!4
\caption{\centerline{\bf F}} }}
\bigskip
```

Komenda `\hgap` wykorzystana powyżej tworzy węższą kolumnę określonego rozmiaru, może być stosowana do składu podwójnych kresek pionowych. I na koniec coś zabawnego:

1	1
2	2
3	3
4	4

Możesz wyeksponować coś takiego za pomocą kresek różnej grubości.

Plik wejściowy tego fragmentu ma postać:

```
\centerline{\othickness=0pt \nexttovr=,1in \abovehr=,1in \belowhr=0in
\table{
\large{2,2in}{Możesz wyeksponować coś takiego za pomocą kresek różnej
grubości.}\global\othickness=3pt
\caption{}}
\othickness=,4pt %reset komendy \othickness
\quad\quad
{\othickness=1pt \ithick=1pt
\table{
1 \global\othickness=0pt \rr 2 \global\othickness=1pt \rr
3 \global\othickness=0pt \rr 4 \global\othickness=1pt
\caption{} }}
\othickness=,4pt %reset komendy \othickness
\bigskip
```

Zamiast wykrzyknika wykorzystana tu została komenda `\centerline`.

`\tabskip``\tabskip * 50`

Ustawia odległość między kolumnami wyrównania. Normalnie wynosi ona 0, ale może być zmieniona. Na przykład `\tabskip=3cm` rozsuwa kolumny (w wyrównaniu `\halign`) na odległość 3 cm.

tabulowanie

Jest bardzo łatwo składać tabele i umieszczać tekst w różnych pozycjach na stronie.

Plik wejściowy dwóch ostatnich wierszy ma postać:

```
{\settabs 4\columns
\+ Jest & bardzo łatwo & składać & tabele \cr
\+ i umieszczać & tekst w różnych & pozycjach & na stronie.\cr}
```

W tabeli 4 w przykładzie VI pokazano różne rodzaje wyrównań. Patrz też `\halign`.

T_EX

Cała książka poświęcona jest wyjaśnieniu tego właśnie pojęcia. Logo T_EX składa komenda `\TeX`, zdefiniowana następująco:

```
\def\TeX{T\kern-,1667em \lower,5ex\hbox{E}\kern-,125em X}
```

Patrz także Plain T_EX i komendy podstawowe.

`\textindent 10–15`

Umieszcza wybrany symbol w miejscu wcięcia akapitu. Patrz, jak stworzono punktory (●) we wstępie.

`$_\textstyle$ * 40–41, 60–63`

Określa krój pisma tekstu w wyrażeniach matematycznych. Komenda ta może być wykorzystana do nadpisania innych stylów, jeśli to konieczne (np. w matematycznych wyrażeniach eksponowanych).

`\the * 2, 80–81, 88–89`

Udostępnia zawartość różnych rejestrów T_EX-a.

`\thinspace 10–11, 24–29, 86–87`

Wstawia odstęp takiej[] wielkości (`\takiej[\thinspace]wielkości`). Komenda ta jest zdefiniowana w Plain T_EX-u następująco:

```
\def\thinspace{\kern ,16667em }
```

`\time *`

Za pomocą tej komendy można uzyskać informację dotyczącą aktualnego czasu, np. teraz jest 905 minut po północy (`\number\time\ minut`). Patrz także `\year`.

`to 74–75`

Patrz słowa kluczowe.

`\tolerance * 80`

Parametr o wartościach z zakresu od 0 do 10000, określający, jak dużo pustego odstępu T_EX może tolerować. Jego wartość domyślna w Plain T_EX-u wynosi 200. Jest on bardzo restrykcyjny w przypadku rozmiarów poziomych węższych niż domyślny rozmiar Plain T_EX-a (6,5 cala) i może się zmieniać. Patrz definicja komend tworzących trójkolumnowy plik wyjściowy w przykładzie XVII.

\topglue

Komenda wstawiająca odstęp u góry strony, np. `\topglue2in` zostawi 2 cale.

\topinsert

Działa podobnie jak `\midinsert`, z tym że najpierw próbuje umieścić materiał u góry bieżącej strony; jeśli nie znajdzie miejsca, umieszcza materiał u góry następnej wolnej strony.

\topmark *

Patrz `\mark`.

\topskip * 80

Ilość odstępu pozostawiona pomiędzy pierwszą linią podstawową strony a szczytem strony. Wartość domyślna tego parametru w Plain TeX-u wynosi 10 punktów.

true [viii–ix](#), [xiv–2](#), [6–7](#), [20–23](#), [44–45](#), [58–59](#), [66–69](#), [74–75](#), [80](#)

Wykorzystywana do określania wielkości niezależnych od powiększenia. Patrz słowa kluczowe.

tryby pracy

Kiedy TeX przetwarza plik, znajduje się w dowolnym z sześciu możliwych trybów pracy. Określony tryb pracy determinuje reakcję TeX-a na odpowiednie komendy. Te tryby to:

- *Poziomy*: przy składzie poziomym — wewnątrz akapitu. Patrz akapity.
- *Ograniczony poziomy*: podczas składu `\hbox`-ów.
- *Pionowy*: przy składzie pionowym — budowanie strony (TeX znajduje się między akapitami).
- *Wewnętrzny pionowy*: przy składzie `\vbox`-ów.
- *Matematyczny*: przy składzie wzorów wewnątrz akapitu.
- *Matematyczny eksponowany*: skład wyrażeń eksponowanych.

\tt [6–9](#)

Składa maszynowy krój pisma (`{\tt maszynowy krój pisma}`).

ułamki [28–29](#), [38–43](#), [52–55](#), [60–63](#)

Patrz przykład II oraz matematyka: efekty specjalne.

\underbar [xiv–1](#)

Podkreśla `tekst` (`\underbar{tekst}`). Podkreślanie nie jest często stosowane w składzie wysokiej jakości: zmiana kroju pisma jest bardziej preferowanym sposobem zwrócenia na coś uwagi. (Zwróć uwagę, jak podkreślenie przecina literę „p”.)

\$_\underbrace\$ [62–63](#)

Umieszcza nawias nad wzorem; patrz równanie 9 w przykładzie IX.

\$_\underline\$ *

Podkreśla wzory, np. `$_\underline{e^{i\pi}+1=0}$` składa $e^{i\pi} + 1 = 0$.

\uppercase

\uppercase *

Składa tekst WERSALIKAMI; plik wejściowy tego fragmentu ma postać:

```
Składa tekst \uppercase{wersalikami}; plik wejściowy...
```

\vadjust *

Patrz puste odstępy i łamanie stron.

\valign *

Komenda działająca podobnie jak `\halign`, z tym że dotyczy wyrównania pionowego. Na przykład

JedenDwa

tak nie

powstało z następującego pliku wejściowego:

```
\smallskip
\valign{\hbox{\strut\sl #} \vfil & \vfil \hbox{\strut #} \cr
        Jeden&tak\cr
        Dwa &nie\cr}
\smallskip
```

\vbadness * 80

Komenda będąca pionowym analogiem `\hbadness`. Każdy `\vbox`, którego kara (*badness*) przekracza określoną wartość `\vbadness`, będzie wypisany przy przetwarzaniu pliku przez \TeX -a. Tak jak i `\hbadness`, wartość `\vbadness` nie ma wpływu na sposób składu; określa, na które pudełka należy zwrócić szczególną uwagę. W Plain \TeX -u `\vbadness` ma wartość domyślną 1000 (może przyjmować wartości od 0 do 10000). Można także stosować większe wartości, np. przy składzie wielokolumnowym. Patrz uwagi końcowe, komendy dotyczące podwójnych kolumn w uwagach.

\vbox * 44–45, 48–51, 66–69, 74–87

Pudełko pionowe. Patrz pudełka. Przykład XV przedstawia sposób umieszczania materiału w takich pudełkach i sposób ich łączenia. Kiedy \TeX składa pudełko pionowe przy użyciu komendy `\vbox`, wybiera linię podstawową ostatniego wiersza jako linię podstawową całego pudełka. Kiedy składa pudełka, wyrównuje je wzdłuż tych linii.

`\vbox` przyjmuje domyślnie wysokość odpowiadającą zawartości pudełka i szerokość odpowiadającą bieżącej wartości `\hsize`. Inne rozmiary należy wyspecyfikować. Na przykład

Jeden

Jeden

Jeden

Dwa

Dwa

Dwa

uzyskano z wpisu:

```
\medskip
\hbox{\vbox to 1in{\hsize 1 in Jeden\vfil Dwa}
      \vbox to ,5in{\hsize ,6in Jeden \vfil Dwa}
      \vbox{\hsize ,6in Jeden \vfil Dwa} }
\smallskip
```

Jak to widać w pliku wejściowym, nowa szerokość kolumny może być wyraźnie wyspecyfikowana wewnątrz `\vbox`-a, rozmiar pionowy zaś może być narzucony z klawiatury. `\vbox` bez wyraźnie wyspecyfikowanego rozmiaru pionowego może jednak istnieć, np. pudełko o jeden cal większe od zawartości: `\vbox spread 1in{}`.

`\vcenter` *

Komenda z trybu matematycznego, podobna do `\vbox` czy `\vtop`, z tym że pionowo wyśrodkowuje składane pudełka. Na przykład

Jeden	Trzy
Dwa	Cztery
	Pięć

uzyskano z:

```
\medskip
\vbox{$\vcenter{\hsize 1in Jeden\hfil\break Dwa}$
      $\vcenter{\hsize 1 in Trzy\hfil\break Cztery\hfil\break Pięć}$}
\smallskip
```

Jeśli zamiast `\vcenter` wykorzystana zostanie komenda `\vbox` (czy `\vtop`), pudełka będą wyrównane wzdłuż dolnych linii podstawowych. Patrz opisy `\vbox` i `\vtop`.

`\vfil` * [vi-ix](#), [8-9](#), [18-19](#), [32-33](#), [74-75](#)

Wypełnia stronę pionowym justunkiem. Patrz `\null`.

`\vfill` * [74-75](#)

Komenda działająca podobnie do `\vfil`; jest jednak silniejsza. Patrz opisy `\hfil` i `\hfill`.

`\vfilneg` *

Przeciwnieństwo `\vfil`. Patrz `\hfilneg`.

`\vfootnote` [40-41](#)

Zwykle przypisy odnoszą się do głównych części tekstu. (Patrz przykład II i VI.) Komenda `\footnote` znaczy tekst, który ma być w przypisie, i używa tego samego znacznika w stopce. Dobór znacznika jest sprawą indywidualną. Komenda ta nie działa niestety wewnątrz skomplikowanych konstrukcji (jak wzory wewnątrz wzorów) lub wewnątrz wstawki (jak np. `\topinsert`). `\vfootnote` pozwala obejść takie sytuacje, dostarcza mechanizmu, który nie łączy przypisu z danym fragmentem tekstu. Umieszcza się komendę w dowolnym miejscu na stronie, gdzie znajduje się związany tekst i niezależnie znaczy tekst tym samym znacznikiem co `\vfootnote`. Przypis w przykładzie V został umieszczony właśnie w taki sposób.

`\vfuZZ` *

Określa, jak bardzo `\vbox` może „sterczeć”, zanim T_EX wyświetli wiadomość o błędzie (*overfull vbox*). Wartość ta w Plain T_EX-u wynosi `0,1pt`. Patrz `\hfuzz`.

`\vglue`

Wstawia odstęp pionowy w dowolnym miejscu, nawet u góry strony. Na przykład odstęp 2-calowy otrzymujemy po wpisaniu `\vglue 2in`. Patrz `\vskip`.

`\voffset``\voffset` * [viii-ix](#), [xiv-1](#)

Kontroluje pionowe położenie strony. Plik wejściowy początku wstępu pokazuje, jak ustawić `\voffset`. Patrz także `\hoffset`.

`\vphantom` [60-63](#), [76](#)

`\vphantom{tekst}` liczy wysokość i głębokość tekstu, ale przypisuje mu zerową szerokość i nie drukuje go. Patrz `\hphantom` i `\phantom`.

`\vrule` * [vi-vii](#), [48-49](#), [76-83](#)

Rysuje kreski pionowe. Patrz `kreska` (rule).

`\vsize` * [viii-ix](#), [xiv-1](#)

Kontroluje pionowy rozmiar strony i pudełek na stronie. Określenie nowej wartości `\vsize` pudełka nie będzie miało wpływu na `\vsize` strony wyjściowej. Domyślny rozmiar pionowy strony w Plain TeX-u wynosi 8,9 cali. Może on być zmieniony: patrz plik wejściowy str. 1. W powiększonym dokumencie należy przy określaniu `\vsize` użyć słowa kluczowego `true`. Patrz jednostki.

`\vskip` * [viii-ix](#), [xiv-1](#), [13](#), [20-21](#), [36-37](#), [58-59](#), [66-67](#), [76-89](#)

Wstawia pionowy odstęp określonego rozmiaru. Komenda ta nie działa u góry strony. `\vskip` kończy aktualny akapit, tak więc odstęp międzyakapitowy (aktualna wartość `\parskip`) jest dodawany do żadanego odstępu. Patrz także `\vglue` i `\relax`.

`\vsplit` * [80](#)

Dzieli `\vbox` na pudełka mniejszej wysokości. Wykorzystywana przy składzie wielokolumnowym.

`\vss` *

Elastyczność pionowa kleju. Patrz `\hss`.

`\vtop` *

Podobna do `\vbox`, z tym że górna linia bazowa jest traktowana jako linia podstawowa pudełka. Na przykład

Jeden	Trzy
Dwa	Cztery
	Pięć

uzyskano z wpisu:

```
\medskip
\hbox{\vtop{\hsize 1in Jeden\hfil\break Dwa}
      \vtop{\hsize 1in Trzy\hfil\break Cztery\hfil\break Pięć}}
\smallskip
```

wcięcia

Patrz `\indent` i akapity.

\wd *

Drukuje szerokość pudełek tworzonych przez T_EX-a. Na przykład materiał bieżącej strony jest umieszczony w rejestrze `\box255`. Szerokość tego pudełka można uzyskać komendą `\wd`. W tym momencie jest to 0.0pt. Wartość tę uzyskano za pomocą komendy `\the\wd255`.

wiązania (ties) 2, 22, 26, 34

Patrz podział wierszy.

wstawki

Patrz przykłady III i VI.

wykresy

Na rysunki czy wykresy (schematy, diagramy) można zostawić puste odstępy, jak to pokazano w przykładzie III, i domontować je w druku. Nie jest łatwo stworzyć diagram w T_EX-u (choć niektóre fonty czasem to umożliwiają), istnieje jednak komenda `\special`, pozwalająca na import grafiki z innych (typowo graficznych) programów. W każdym systemie wygląda to inaczej, więc o szczegóły należy pytać administratorów systemów lokalnych.

wyliczenia

Patrz `\item`, `\itemitem`, spis treści, cytowana literatura i przykład IV.

wymiary

Patrz jednostki.

wyrównywanie 26–27, 40–51

Patrz macierze oraz tabele.

wyrównywanie równań 38

Patrz przykłady V, VII i IX oraz objaśnienia komend `\eqalign`, `\eqalignno` i `\leqalignno`.

wysokość (height) vi–vii, 12–13, 34–37, 48–49, 66–69, 76–85

Odnosi się do wysokości pudełek lub kresek ponad ich liniami podstawowymi. Patrz pudełka i `\ht`. Jest to także słowo kluczowe odnoszące się do wysokości kresek.

wyśrodkowanie vi–vii, 6–9, 20–21, 44–45

Aby wyśrodkować materiał pionowo na stronie, należy przed i za nim umieścić komendę `\vfil`. Na końcu strony umieszcza się komendę `\eject`, aby upewnić się, że inne silniejsze komendy dotyczące odstępów nie odsuną materiału do góry. Patrz także `\null`.

Do wyśrodkowania poziomego można stosować kilka metod, np. do wyśrodkowania tekstu w wierszu instrukcję `\centerline`. W przypadku wyrażeń matematycznych stosuje się komendę `\centerline` i znak `$` do włączenia/wyłączenia trybu matematycznego lub podwójny znak `$$` wokół wyrażenia. W tym przypadku wyrażenie jest wyśrodkowane, wyeksponowane w większym formacie oraz dołożone są dodatkowe odstępy między wyrażeniem a otaczającym tekstem. Patrz równania 1–3 w przykładzie II. Styl eksponowany może być wykorzystany do wyśrodkowania tekstu `$$\vbox{tekst}$$`, jak to zrobiono w tabeli 3 (przykład VI).

`\year``\year *`

Mamy teraz rok 2003. Zdanie takie uzyskano za pomocą komendy `\number\year`.
 Wszystkie komendy dotyczące daty i czasu można użyć w następujący sposób:

```
\newcount\mins \newcount\hours \hours=\time \mins=\time
\def\now{\divide\hours by 60 \multiply\hours by60 \advance\mins by-\hours
\divide\hours by60
\ifnum\hours>12 \advance\hours by-12
\number\hours:\ifnum\mins<10 0\fi\number\mins\ P.M.\else
\number\hours:\ifnum\mins<10 0\fi\number\mins\ A.M.\fi}
\def\today {\monthname\ \number\day, \number\year}
```

Te komendy pozwalają z wpisu:

Aby być precyzyjnym, jest `\now\ today`.

uzyskać w składzie stwierdzenie:

Aby być precyzyjnym, jest 3:05 P.M. marzec 17, 2003 roku.

znaki specjalne [6-19](#), [24-25](#), [40-41](#)

œ <code>\oe</code>	æ <code>\ae</code>	å <code>\aa</code>	ø <code>\o</code>	ı <code>\l</code>
Œ <code>\OE</code>	Æ <code>\AE</code>	Å <code>\AA</code>	Ø <code>\O</code>	Ł <code>\L</code>
† <code>\dag</code>	‡ <code>\ddag</code>	§ <code>\S</code>	¶ <code>\P</code>	ß <code>\ss</code>
ı <code>\i</code>	ı <code>\j</code>	... <code>\dots</code>	© <code>\copyright</code>	

Istnieje także symbol £ składany komendą `{\it\}$}`


```

1 % Ten plik jest krótki. Ustawia komendę "\literal", przywołuje kolejne pliki
2 % formatujące i wyświetla ich zawartość. Interesujący jest fakt, że jeden
3 % z tych plików (TBEINPMA) został wykorzystany do sformatowania tej strony.
4
5 \hsize 5,5 true in \vsize 9 true in \voffset ,1 true in
6 \nopagenumbers \pageno=161 \raggedbottom
7 \font\titl = plssdc10 scaled 2986 % tytuł
8 \font\bsl = plsl12 % podtytuł
9 \font\hl = plti12 % nagłówek
10 \font\pn = plbx12 % numeracja stron
11 \font\inpfile = pltt8 % pliki formatujące
12
13 \def\heading {{\hl Uwagi końcowe (nie dla początkujących)}}
14 \def\Page1 {\llap{\hbox to 1,125 true in{\pn\the\pageno\hfil}}\heading\hfil}
15 \def\Pager {\hfil\heading\rlap{\hbox to 1,125 true in{\hfil\pn\the\pageno}}}
16
17 \headline{\ifnum\pageno=161 {\hfil} \else
18 \vbox{\line{\ifodd\pageno \bsl\botmark \Pager \else \Page1 \bsl\botmark \fi}
19 \smallskip \hrule width\hsize}\fi}
20
21 \output={\ifodd\pageno \hoffset ,125 true in \else \hoffset ,875 true in \fi
22 \plainoutput} % '\plainoutput' to standardowa komenda Plain \TeX-a
23
24 \rightline{\title Uwagi końcowe} \vskip 10pt
25 \rightline{\bsl Plik wejściowy do pliku wejściowego} \vskip 10pt
26 \rightline{\bsl (czyli jak złożyć książkę).} \vskip 2,2 true in
27 \vskip-2\baselineskip \vskip-20pt % w celu zrównoważenia podtytułu
28
29 \noindent{\bf Poniższy tekst nie jest przeznaczony dla początkujących.}
30 Przedstawia pliki, które były wykorzystane do stworzenia formatu stron
31 wejściowych, dodatku, itd. Instrukcja, jak wykorzystać takie pliki, znajduje
32 się w dodatku pod hasłem {\tt\string\input}. \smallskip \TeX\ ma możliwości
33 dużo bardziej wyszukanego formatowania niż skład wykorzystany w tej książce.
34 Nie znajdziesz tu takich zastosowań jak automatyczne indeksowanie
35 i odwołania. Pomimo tych wyłączeń, pliki przedstawione na kilku następnych
36 stronach nie są łatwe do czytania. Starałem się umieścić komentarz obok
37 każdej nowej komendy, wyjaśnienie, w jakim celu jest umieszczona, sposób,
38 w jaki działa. Niestety, jednak formatowanie książki --- nawet dla nowicjuszy
39 --- wymaga zagadnień, które przekraczają zwykły poziom dla początkujących.
40
41 \def\nf #1{\bigbreak\hrule height 1pt\smallskip{\noindent\bf #1}\smallskip\hrule
42 \bigskip \mark{#1}} % w celu wprowadzenia poniżej nowych plików.
43
44 % Komendy w następnych wierszach są potrzebne przy wydruku zawartości plików.
45 \def\cc {\catcode}
46 {\cc'\^^M=\active \gdef\loosenolines{\cc'\^^M=\active \def^^M{\null\par}}}
47 \def\literal{\begingroup \cc'\=12 \cc'\{=12 \cc'\}=12 \cc'\$=12 \cc'\&=12
48 \cc'\#=12 \cc'\%=12 \cc'\~=12 \cc'\_ =12 \cc'\ " =12 \cc'\^ =12
49 \obeyspaces \loosenolines \baselineskip=9pt \tolerance=10000 \inpfile}
50 {\obeyspaces\gdef {\hglue,5em\relax}}% UWAGA: ,,spacja'' staje się
51 % znakiem sterującym,
52 \cc'\/=0 % znak / nowym symbolem
53 % początku sekwencji sterujących.
54 \cc'\+=10 % Do wykorzystania znaku ,,+" jako ,,spacji'', dopóki jest ona
55 % znakiem sterującym w komendzie \literal.
56
57 % Każdy z plików jest teraz zatytułowany i dołączony:
58 \nf{Plik użytkowy} \literal/input+tbeutili/endgroup \bigskip
59 \nf{Format pliku wejściowego}\literal/input+tbeinpma/endgroup \bigskip
60 \nf{Format pliku wyjściowego}\literal/input+tbeoutma/endgroup \bigskip
61 \nf{Przetwarzanie pliku wyjściowego}\literal/input+tbemain1/endgroup \bigskip
62 \nf{Format dodatku} \literal/input+tbeappnd/endgroup \bigskip
63 \nf{Strony początkowe} \literal/input+tbefmatt/endgroup \bigskip
64 \hrule height1pt \bigskip
65 \centerline{\bf KONIEC}
66 \bye

```

Uwagi końcowe

*Plik wejściowy do pliku wejściowego
(czyli jak złożyć tę książkę).*

Poniższy tekst nie jest przeznaczony dla początkujących. Przedstawia pliki, które były wykorzystane do stworzenia formatu stron wejściowych, dodatku, itd. Instrukcja, jak wykorzystać takie pliki, znajduje się w dodatku pod hasłem `\input`.

\TeX ma możliwości dużo bardziej wyszukanego formatowania niż skład wykorzystany w tej książce. Nie znajdziesz tu takich zastosowań jak automatyczne indeksowanie i odwołania. Pomimo tych wyłączeń, pliki przedstawione na kilku następnych stronach nie są łatwe do czytania. Staralem się umieścić komentarz obok każdej nowej komendy, wyjaśnienie, w jakim celu jest umieszczona, sposób, w jaki działa. Niestety, jednak formatowanie książki — nawet dla nowicjuszy — wymaga zagadnień, które przekraczają zwykły poziom dla początkujących.

Plik użytkowy

```
% OTO PLIK ,,TBEUTILI''. ZAWIERA DEFINICJĘ KOMENDY ,,\ignore'', WYKORZYSTANEJ
% W POZOSTAŁYCH PLIKACH FORMATUJĄCYCH PODRĘCZNIKA

\def\ignore{\begingroup \count255=0
  \loop \catcode\count255=14 %wszystkie znaki stają się znakami komentarza
  \advance \count255 by1 \ifnum\count255<128
  \repeat \catcode'\!=0 }
{\catcode'\!=0 !gdef!E{!endgroup}} %znak ! staje się znakiem sterującym

% KOMENDA TA ZOSTAŁA OMÓWIONA W DODATKU POD HASŁEM ,,komentarze''.
```

Format pliku wejściowego

```
% OTO PLIK ,,TBEINPMA''. ZAWIERA ,,MAKRA'' (CZYLI KOMENDY) NAPISANE W CELU
% ZŁOŻENIA STRON WEJŚCIOWYCH PODRĘCZNIKA ,,TeX\ W PRZYKŁADACH''.

\input tbeutili

\input tbe4pdf.tex % modyfikacja 2003; d(JK)

%***** NAZWY FONTÓW *****
\font\pn = plbx10 scaled\magstep1 % numeracja stron wejściowych
\font\blare = plbx10 scaled\magstephalf % napis "UWAGI"
\font\ssn = plss9 % numeracja wierszy
\font\standout = plcsc10 % tekst wyróżniony w uwagach
\font\emphtt = pltt10 % teksty ,,verbatim''

%Fonty wykorzystane w uwagach.
\font\ninerm=plr9 \font\ninesl=plsl9 \font\nineit=plti9
\font\ninett=pltt9 \font\ninebf=plbx9 \font\ninemi=plmi9
\font\ninesy=plsy9

\def\ninepoint{\let\rm=\ninerm \let\sl=\ninesl \let\it=\nineit
```



```

\let\tt=\ninett \let\bf=\ninebf \let\mit=\ninemi \let\cal=\ninesy
\baselineskip=11pt \rm}

%***** OTWARZANIE PLIKU WEJŚCIOWEGO *****
\def\cc{\catcode} % skrót wykorzystywany przy zmianie kategorii znaków

% Następną komendą otwiera wejściowy „verbatim” przez wyłączenie specjalnej
% kategorii znaków i przypisanie kodu 12 wszystkim komendom znakowym.
\def\literal {\begingroup \cc\=12 \cc\{=12 \cc\}=12 \cc\%=12
\cc\&=12 \cc\#=12 \cc\%=12 \cc\^=12 \cc\_=12 \cc\^=12
\cc\‘=\active \obeyspaces \inboxfont} % „\obeyspaces” powoduje, że odstęp
% staje się znakiem sterującym.
\obeyspaces\gdef {\hglue,5em\relax}} % odstęp międzyznakowy

\cc\‘=\active \gdef{\relax\lq}} % zablokowanie pewnych ligatur.

%***** SKŁAD PUDEŁEK WEJŚCIOWYCH *****
% ‘\autoln’, poniżej składa kolumnę liczb, która jest wykorzystana do
% numerowania wierszy w pliku wejściowym. Komenda posiada argument, będący
% maksymalną liczbą wierszy w pudełku (policzoną gdzie indziej). Liczby są
% składane krojem \ssn. Wiersze są odsunięte na odległość co najmniej rozmiaru
% „,y’” za pomocą komendy \holdapart --- „,y’” ma dużą wysokość i głębokość.

\def\inboxfont{\emphtt} % font stosowany w pudełkach wejściowych
\def\linesp{\baselineskip=11pt} % interlinia tekstów ww. pudełek

\def\holdapart{\vphantom{\inboxfont(y)}
\newcount\lno
\def\autoln #1{\vbox{\hsize12,5pt \linesp \ssn \lno=1
\loop \holdapart\hfill\hblue{number\lno}\kern3pt\ifnum\lno<#1\advance\lno by1
%
% modyfikacja 2003; d(JK)
\break\repeat
\vskip\innerspace\vskip\th}}

% - - - - -
% komenda „\inbox” składa pudełka wejściowe. Pierwszy argument to poziomy
% rozmiar pudełka, drugi -- jego zawartość. Najpierw wprowadzono nazwy nowych
% zmiennych oraz nowych pudełek, które będą składać tekst pliku wejściowego.

\newdimen\innerspace \innerspace=4,1pt % odstęp od krawędzi wewnątrz pudełka
\newdimen\th \th=,4pt % grubość obwódki pudełka
\newbox\inp % to stanie się pudełkiem wejściowym
\newcount\maxlno % maksymalna liczba # wierszy
\lineskiplimit=-2pt % dołożona w polskim wyd. (ujemny odstęp dla górnych
% akcentów polskich liter --- przyp. tłum.

\def\inpbox #1#2{\setbox\inp=
\vbox{\hsize#1 \linesp \maxlno=1 \noindent \holdapart #2\holdapart}
\hbox{\autoln{\maxlno}\vbox{\hrule height\th depth0pt
\hbox{\vrule width\th\hskip\innerspace
\vbox{\vskip\innerspace \unvbox\inp \vskip\innerspace}
\hskip,1pt\vrule width\th}\hrule height\th depth0pt}}}

% w celu powiększenia \maxlno o 1 przy każdym przywołaniu
\def\track {\null\hfil\break\global\advance\maxlno by1}

% komenda ta powoduje, że znak końca wiersza staje się znakiem sterującym
% i za pomocą komendy \track zlicza wiersze:
\cc\^M=\active \gdef\countlines {\cc\^M=\active \def^M{\track} }}

% znak „,|’” będzie teraz znakiem sterującym, tak więc może zostać wykorzystany
% jako specyficzna komenda. Komenda ta uaktywnia \literal, w momencie pierwszego
% pojawienia się, kończy zaś grupę przy drugim pojawieniu się (wykorzystując
% znak „,|’” --- znak wyjścia) i wyłącza komendę \literal.
% Znak „,|’” również przywołuje komendę \inpbox. Jest jeszcze jeden znak
% specjalny: „,*’”, spełniający funkcję znaku komentarza.

\newdimen\boxwidth % wartość będzie przypisana później
\cc\|= \active
\def|{\literal \countlines \cc\@=0 \cc\*=14 \scrbox}
\cc\@=0 \cc\|=12
\gdef@scrbox#1|{@inpbox{\boxwidth}{#1}@endgroup}}

```



```

\cc\'"=12 \onward}

%***** PARAMETRY STRONY *****
\nopagenumbers      % zakaz numerowania stron
\parindent=0pt      % ustawienie zerowego wcięcia akapitu
\parskip=0pt        % ustawienie zerowego odstępu między akapitami
\ninepoint          % określenie stopnia pisma (9 pt)

% Ustawienia poziome
\newdimen\strip \strip=,3125 true in % \strip --- pozostawienie odstępu
\boxwidth=5,8125 true in % szerokość pudełek wejściowych
\hsize=\boxwidth \advance\hsize by\strip
\newdimen\rightmargin \rightmargin=1,125 true in
\hoffset 7,5 true in
\advance\hoffset by-\rightmargin \advance\hoffset by-\hsize
\newdimen\colwidth \colwidth=\boxwidth
\divide\strip by2 \advance\colwidth by\strip
\divide\colwidth by2 % szerokość kolumny dla tekstu "UWAGI"

% Ustawienia pionowe
\vszize 9,625 true in \voffset -,3125 true in

%***** PRZYKŁADOWY PLIK ŹRÓDŁOWY STRONY WEJŚCIOWEJ *****
\ignore
% PLIK ŹRÓDŁOWY STRONY 32
\pageinsert \centerline{\bf Rysunek 1} \vfil
\item{\bf 1a.}{Dwuwymiarowa przestrzeń Minkowskiego;  $ds^2 = - dt^2 - dx^2$ .
 $\mathcal{S}$  jest powierzchnią  $t=1$ .  $D^{\pm}(\mathcal{S})=I^{\pm}(\mathcal{S})$ .
Pokazane są także zbiory  $I^{\pm}(p)$  dla pewnego  $p$  w  $\mathcal{M}$ .} \vfil
\item{\bf 1b.}{Kolejny przykład dwuwymiarowej przestrzeni Minkowskiego;
 $\mathcal{S}$  jest zadana przez  $t=-\sqrt{1-x^2}$ .
 $D^-(\mathcal{S})=I^-(\mathcal{S})$ , ale  $D^+(\mathcal{S}) \neq I^+(\mathcal{S})$ .
Więc  $H^+(\mathcal{S}) \neq \emptyset$ .} \vfil
\item{\bf 1c.}{Dwuwymiarowa czasoprzestrzeń z cylindrem  $R \times S^1$  jako
rozmaitość;  $ds^2 = -d\psi^2 - dt^2$ ,  $-\infty < t < \infty$ ,
 $0 \leq \psi < 2\pi$ . Tutaj  $t=0$  jest hiperpowierzchnią typu
przeszłego, a  $t=2\pi$  jest przyszłym horyzontem Cauchy'ego. Ten
horyzont Cauchy'ego jest zbiorem zwartym.}
\endinsert |

\bnotes
Oto tekst dotyczący komendy \pageinsert z poprzedniej strony.
Był on wprowadzony jako część głównego dokumentu, z komendą
\endinsert oznaczającą koniec wstawki. \TeX reaguje na te
komendy przez umieszczenie materiału zawartego między nimi na następnej
stronie. Może to być pusta strona, uzyskana w następujący sposób:\ee
"\pageinsert"\ee
"\vfil"\ee
"\endinsert"\ee
lub dowolna wstawka, jak w tym miejscu.\ee
Pomiędzy nagłówkami rysunków pozostawiono puste miejsce na grafikę "\vfil",
która została wykonana w innym programie [w tym przypadku w programie
MetaPost --- przyp. tłum.]. Możliwy jest elektroniczny „import” plików
graficznych z innych źródeł i umieszczanie ich w tekście np. za pomocą
komendy "\special" (komenda ta nie jest interpretowana przez \TeX-a, lecz
umieszczana w pliku *.dvi wraz z argumentem, który jest znakiem sterującym
czytelny dla urządzenia drukującego; szczegóły można znaleźć w literaturze
[1]--- przyp. tłum.)\ee
\bo{9} W tym miejscu pokazano jak składać pierwiastek "\sqrt" obejmujący
dłuższe wyrażenie.\ee
\bo{13} "$\psi$" składa  $\psi$ , a  $-\infty$  składa  $-\infty$ .
Komenda "\", tworzy odstęp, małą spację pomiędzy  $d\psi$  i  $dt$ .
\bo{15} "$xleq$" składa  $xleq$ .\ee
\separator
\centerline{\[Komentarz]}\ee
Znaki "$" określają, które z komend pracują wyłącznie w trybie
matematycznym.\ee
\enotes
!E

```

Format pliku wyjściowego

```
% TO JEST PLIK ,,TBEOUTMA''. ZAWIERA ON KOMENDY FORMATUJĄCE WYKORZYSTANE DO
% SKŁADU STRON WYJŚCIOWYCH PODRĘCZNIKA (PATRZ --- KOMENDA ,,input'', STR. 60.)

\magnification=\magstep1
\font\ttitle=plssdc10 scaled 2488
\baselineskip 22 true pt

\hsz 5,5 true in \hoffset ,125 true in
\vsz 8,5 true in \voffset ,1 true in
\nopagenumbers

\def\newpageno {{\multiply\pageno by2 \advance\pageno by-1
\rlap{\hbox to 1,125 true in{\hfil\bf\the\pageno}}}}
```

Przetwarzanie pliku wyjściowego

```
% PLIK TEN JEST WYKORZYSTANY JAKO ,,PROCESOR WYJŚCIOWY''. Przetwarza tekst
% stron ,,wejściowych'' --- dzieli go na części dotyczące uwag, pudełek
% z plikiem wejściowym, itd. Tworzy kolejne strony wyjściowe. (Uwagi są dodawane
% po próbnym przebiegu i złamaniu stron.) Ten szczególny procesor działa na
% jeden z dwóch głównych plików (str. xiv-59 podręcznika). Są jeszcze inne
% procesory przeznaczone do przetwarzania pozostałych plików tej książki.

% Znak ,,@'' staje się znakiem komentarza; znak ,,*' znakiem komentarza.
\catcode'\@=14 \catcode'\*=9

\catcode'\|= \active
\def\skipstuff {\begingroup \count255=0
\loop \catcode\count255=14
\advance\count255 by1 \ifnum\count255<128
\repeat
\catcode'\|= \active \catcode'\*=9 \catcode'\|= \active
\catcode'\.=0}
\def|\{\skipstuff \let|= \endgroup }
% Tak więc każde nieparzyste pojawienie się znaku | rozpoczyna
% pomijanie materiału, parzyste --- zatrzymuje pomijanie.

{\catcode'\.=0 .gdef.E{\endgroup }} %

{\catcode'\|= \active \gdef,{\input epilogue }}
% Pojedynczy znak przecinka znajdzie się w pliku wejściowym poniżej:

|,
```

Format dodatku

```
% TO JEST PLIK ,,TBEAPPND''. ZAWIERA KOMENDY WYKORZYSTANE DO FORMATOWANIA
% DODATKU

\input tbeutil

%***** NAZWY FONTÓW *****
\font\ttitle = plssdc10 scaled 2986 % tytuł sekcji
\font\hfont = plti12 % nagłówek dodatku
\font\pn = plbx12 % numery stron
\font\htopic = plss10 scaled\magstephalf % hasła na 1 stronie
\font\start = plr10 scaled\magstephalf % pierwsza strona
\font\emph = plsl10 scaled\magstephalf % emfaza ze strony 1
\font\bigsl = plsl10 scaled\magstep1 % podtytuły
\font\ntt = pltt9 % fragmenty pliku wejściowego
\font\topic = plss10 % odwołania do haseł
\font\bt = pltt10 scaled\magstephalf % komendy (duży rozmiar)
\font\topichd = plssdc10 scaled\magstephalf % hasła (duży rozmiar)

%***** USTAWIENIA DOTYCZĄCE NAGŁÓWKA I NUMERACJI STRON *****
% Najpierw definiujemy komendę pomocniczą do komendy ,,raggedright''
\def\raggedleft{\leftskip=0pt plus2em \parfillskip=0pt
```

```

\spaceskip=,3333em \xspaceskip=,5em\relax}
% UWAGA: ,, \spaceskip'' i ,, \xspaceskip'' pozwalają kontrolować odstępy
% między wyrazami

\newcount\pno \pno=0 % ,, \pno'' jest wskaźnikiem, potrzebnym później
\def\header{\hfont Dodatek: skorowidz i słownik}}
\def\startat#1#2{\pageno=#1 \global\pno=#2
\headline{\ifnum\pno=1 \hfil \else
\ifodd\pageno
\hfil \header \rlap{\hskip ,1in\vtop{\hsize 1,025in
\pretolerance=10000 \hbadness=5000
\line{\hfil\pn\folio} \smallskip
\noindent \botmark \raggedleft \endgraf}}
\else
\llap{\vtop{\hsize 1,025in \pretolerance=10000 \hbadness=5000
\line{\pn\folio\hfil} \smallskip
\noindent \firstmark \raggedright
\endgraf}\hskip\shift\hskip,1in}\hskip-\shift \header \hfil
\fi \fi} \ifnum\pno=1 \startingstuff \fi }

\ignore
,, \pno'' jest wskaźnikiem; dla pierwszej strony ma wartość 1, dla pozostałych 0.
,, \ifnum\pno=1 \hfil'' powoduje, że pierwsza strona nie ma nagłówka.
,, \ifodd...\else...\fi'' różnicuje nagłówki stron ,, prawych i ,, lewych''.
,, \hfont'' ustawia krój pisma w nagłówku.
,, \rlap'' umieszcza tekst na prawym marginesie; \llap -- na lewym.
,, \quad'' wstawia odstęp.
,, \vtop'' składa pudełko, którego górna kreska podkreśla napis ,, Dodatek...''
,, \hsize'' ustawia rozmiar poziomy pudełka.
,, \pretolerance=1000'' zapobiega podziałowi wyrazów.
,, \pn'' określa krój pisma; ,, \folio drukuje numer strony.
,, \noindent'' likwiduje wcięcie pierwszego wiersza.
,, \botmark'' i ,, \firstmark'' pierwszy i ostatni znaczonej fragmentu tekstu na str.
,, \raggedright'' wyłączenie prawej justyfikacji.
,, \raggedleft'' wyłączenie lewej justyfikacji.
,, \endgraf'' kończy akapit; \fi kończy komendy warunkowe.
,, \startingstuff'' zdefiniowana na końcu pliku
!E

%***** ADIUSTACJA STRON WYNIKOWYCH *****
\output={\ifodd\pageno \hoffset ,625in \else \hoffset 1,375in \fi
\shipout\vbox{\makeheadline
\moveleft\shift\vbox{\ifnum\pno=0
\hrule width 5,5in height ,3pt \vskip,25in\fi}
\pagebody
\ifnum\pno=1 \vskip,25in \global\pno=0\fi
\makefootline}
\advancepageno }

\ignore
Wyjściowy program standardowy określa końcowe instrukcje budowania stron, po
złożeniu ich głównej zawartości. (Program standardowy pokazany powyżej pomija
niektóre wyszukane cechy, jak np. upewnienie się, że wstawki zostaną
wydrukowane.)
,, \shipout'' komenda wysyłająca strony.
,, \vbox'' ,, wkłada'' materiał ze strony w pudełko pionowe.
,, \makeheadline'' składa nagłówki.
,, \moveleft'' przesuwa pudełko pionowe w lewo, ponieważ wpisy w dodatku sterczą
z lewej strony. Zawartością vbox-a w tym przypadku jest kreska pozioma
i odstęp pionowy 0,25 cala.
,, \pagebody'' wstawia zawartość danej strony.
,, \makefootline'' składa stopkę.
,, \advancepageno'' zwiększa numer strony.
!E

%***** KOMENDY RÓWNIEM PRZYDATNE *****
\def\cc{\catcode} % skrót \catcode
\def\hb {\hfil\break}
\def\[#1]{\begingroup\it\ignorespaces#1\endgroup} % skład kursywą
\def\bl{\item{\$bullet$}} % listy wyliczane
\def\texti{\textindent{\$bullet$}} % akapity wyliczane
\def\par{\endgraf\penalty-500 }

```

```

% UWAGA: ,,penalty'' jest twoim własnym rzutem oka na wewnętrzną pracę \TeX-a:
% wiersze, strony; oszacowaniem kar połączonych z różnymi punktami łamania
% tekstu. Ujemna wartość \penalty informuje o dobrym miejscu do złamania strony.
% Jest to trzon definicji komendy ,,goodbreak''.

%***** KOMENDY ODTWARZAJĄCE INNE KOMENDY *****
{\cc'\^M=\active % patrz plik \tbeinpma
\gdef\loosenolines{\cc'\^M=\active \def^M{\leavevmode\endgraf}}
\def\literal {\begingroup \cc'\=12 \cc'\{=12 \cc'\}=12 \cc'\$=12 \cc'\&=12
\cc'\#=12 \cc'\%=12 \cc'\^=12 \cc'\_ =12 \cc'\^=12 \cc'\*=12 \cc'\@=0
\cc'\='=\active \loosenolines \obeyspaces \tt}%

{\obeyspaces\gdef {\hglue,5em\relax}}

{\cc'\='=\active \gdef{\relax\lq}} % zablokowanie ligatur

\cc'\="=\active
\def"\{\literal\leavevmode\hbox\bgroup\com}
% \leavevmode rozpoczyna nowy akapit, jeśli zachodzi taka potrzeba.

\def\beginliteral{\medskip \literal \cc'\="=12 \ntt %
\parskipOpt \parindent\indsize \baselineskip11pt \thatisit}

{\cc'\@=0 \cc'\=12 @cc'\^M=@active %
@gdef@com#1"{#1@egroup@endgroup} %
@gdef@thatisit^M#1\endliteral{#1@endgroup@smallskip}}

\def\mc #1{{\tt\$ \char'134#1\$}} % zapis komend matematycznych
\def\tc #1{{\tt\char'134#1}} % zapis komend tekstowych
\def\bmc #1{{\btt\$ \char'134#1\$}} % komendy matematyczne (duży rozmiar)
\def\btc #1{{\btt\char'134#1}} % komendy tekstowe (duży rozmiar)

%***** KOMENDY NOWYCH WPISÓW W DODATKU *****
\def\p{\hskip-3pt\ast\hskip10pt\relax} % dla komend podstawowych

\def\entry #1#2{\bigbreak\moveleft\shift\ vbox{#1\quad\rm #2}
\nobreak\vskip\parskip \ignorespaces}

\def\ne #1!#2!{\entry{\topichd #1}{#2}\mark{\topic #1}}
\def\net #1!#2!{\entry{\btc {#1}}{#2}\mark{\tc {#1}}}
\def\nem #1!#2!{\entry{\bmc {#1}}{#2}\mark{\tc {#1}}}

% Każdy z tych wpisów daje możliwość umieszczenia nagłówka hasła na marginesie
% i złożenie hasła i numeru strony właściwym krojem pisma.

%***** WYŚWIETLENIE KOMENDY WRAZ Z EFEKTEM JAKI POWODUJE *****
\def\nr{\hskip-2em\relax\cr} % zlikwidowanie wbudowanego odstępu \quad

\cc'\@=\active % @ jako komenda znakowa
\def#1@{\hfil#1\hfil\ \ &{\tt\$string#1}\quad\hfil&}% dla komend tekstowych

\cc'\*=\active % * jako komenda znakowa
\def#1*{\hfil$#1\hfil\ \ &{\tt\$string#1\$}\quad\hfil&}% w trybie matem.

% eksponowanie tabel
\def\disp #1{\removelastskip \medskip\smallskip \begingroup
\centerline{\vbox{\halign{##}\cr #1\cr}} \endgroup\medbreak\ignorespaces}

% eksponowanie tabel z wbudowanym formatem preambuły
\def\dispf #1{\removelastskip \medskip\smallskip \begingroup
\centerline{\vbox{\halign{##}\cr #1\cr}} \endgroup\medbreak\ignorespaces}

% UWAGA: ,,removelastskip'' likwiduje poprzedzający tabelę \vskip. Zabezpiecza
% to przed pozostawieniem zbyt dużego odstępu przed tabelami.

%***** USTAWIENIA DOTYCZĄCE UKŁADU STRONY *****
\hsize 5in % poziomy rozmiar
\vsize 8,6in \voffset ,1in % pionowy rozmiar
% UWAGA: \hoffset jest wbudowany w standardowy program omówiony powyżej.

\parindent 0pt \parskip 2pt % format akapitu

```

```

\newdimen\indsize \indsize 15pt          % rozmiar ekstra wcięcia
\def\ind{\hskip\indsize\relax}          % wcięcie niektórych akapitów

\newdimen\widthofbook \widthofbook=5,5in % UWAGA: szerokość drukowanych stron

\newdimen\shift \shift=\widthofbook      % ujemne przesunięcie
\advance\shift by -\hsiz                 % wcięcie wpisów w dodatku

\nopagenumbers
\interlinepenalty=200                    % zabronienie łamania strony wewnątrz akapitu

\def\startingstuff{
\rightline{\title Dodatek}              % tytuł dosunięty do prawej
\vskip 10pt
\rightline{\bigsl Skorowidz i słownik.} % podtytuł dosunięty do prawej
\vskip 2,5 true in                       % odstęp 2,5 calowy
\vskip-\baselineskip \vskip-10pt        % w celu skompensowania podtytułu
}

%***** PRZYKŁAD WPISU Z DODATKU *****
\ignore
\ne znaki specjalne!6--19, 24--25, 40--41!
\dispf{@\oe@ @\ae@ @\aa@ @\o@ @\l@ \nr
        @\OE@ @\AE@ @\AA@ @\O@ @\L@ \nr
        @\dag@ @\ddag@ @\S@ @\P@ @\ss@ \nr
        @\i@ @\j@ @\dots@ @\copyright@ \nr}

Istnieje także symbol {\it\$} składany komendą "{\it\$}"
!E

```

Strony początkowe

```

% OTO PLIK ,,TBEFMATT''. ZAWIERA KOMENDY WYKORZYSTANE DO FORMATOWANIA
% STRON POCZĄTKOWYCH PODRĘCZNIKA

\hsiz 4,75 true in \hoffset ,875 true in
\vsiz 8,5 true in \voffset ,1 true in
\nopagenumbers

\font\title = plssdc10 scaled 2986      \font\subtitle = plss10 scaled 2074
\font\author = plssi10 scaled 2488     \font\pub = plr12
\font\ded = plti10 scaled 1440         \font\bsym = plsy10 scaled\magstep1
\font\lrm = plr9 \font\lsl = pls19     \font\lmsy = plsy9
\def\chota{\baselineskip=11pt \let\rm=\lrm \let\sl=\lsl \rm}

\setbox0=\vbox{\subtitle } \dimen0=\ht0 % wykorzystana na stronie tytułowej.

\def\TeX {\hbox{T\kern-.14em\lower,5ex\hbox{E}\kern-.03em X}}
\def\sh {\hskip,6em\relax}
\def\vfitframe #1#2#3#4{\vbox{\hrule height #1pt%
\hbox{\vrule width #1pt\kern #4pt%
\hbox{\kern #2pt\hbox{#3}\kern #2pt}%
\kern #4pt\vrule width #1pt}%
\hrule height 0pt depth #1pt}}
\def\copyr{\leavevmode\hbox{\lmsy\char'15}\llap{\lrm c\kern,3em }}}
\def\forever{\leavevmode\hbox{\bsym\char'15}\llap{%
\raise1,7pt\hbox{\scriptstyle\infty}\thinspace }}}

%***** STRONA PRZEDTYTUŁOWA *****
\vglue ,3in
\centerline{\title \TeX\sh w\sh przykładach} \bigskip
\centerline{\subtitle Podręcznik dla początkujących} \vfil\ejct

%***** STRONA TYTUŁOWA *****
\output={\hoffset,375 true in
\shipout\vfitframe{,4}{0}{\vbox{\pagebody}}{36,14}\advancepageno}
\vglue ,3in
\centerline{\title \TeX\sh w\sh przykładach} \bigskip
\centerline{\subtitle Podręcznik dla początkujących}
\vskip 2,2 true in \vskip-\dimen0 \vskip-\bigskipamount
\centerline{\author Arvind Borde} \vfil

```

```

% logo AP zostało umieszczone przez wydawcę.
\centerline{\pub ACADEMIC PRESS, INC.} \smallskip
\centerline{\pub Harcourt Brace Jovanovich, Publishers} \bigskip
\centerline{Boston\quad SanDiego\quad New York}
\centerline{London\quad Sydney\quad Tokyo\quad Toronto}
\vglue,4in \eject

%***** STRONA WYDAWCY *****
\output={\plainoutput}
\vphantom{\title{}} w celu zapewnienia jednakowych odstępów stron tytułowych
{\rightskip 0pt plus,75in\relax \hbadness 5000 \parindent0pt}
\vfil
This book is printed on acid--free paper. \forever
\bigskip\bigskip\bigskip
Copyright \copyright\ 1992 by Academic Press, Inc. \hfil\break
All rights reserved.\hfil\break
No part of this publication may be reproduced or transmitted in any form or by
any means, electronic or mechanical, including photocopying, recording, or any
information storage and retrieval system, without permission in writing from
the publisher. \bigskip
ACADEMIC PRESS, INC. \hfil\break
1250 Sixth Avenue, San Diego, CA 92101
\bigskip\bigskip\bigskip
{\it United Kingdom Edition published by}\hfil\break
ACADEMIC PRESS LIMITED\hfil\break
24--28 Oval Road, London NW1 7DX
\bigskip\bigskip\bigskip
{\it Library of Congress Cataloging--in--Publication Data}
\bigskip
\vskip11\baselineskip % library of Congress Catalog Data will be pasted here
\bigskip\bigskip\bigskip
{\it Printed in the United States of America}\hfil\break
91\ \ 92\ \ 93\ \ 94\quad 9\quad8\quad7\quad6\quad5\quad4\quad3\quad2\quad1
\bigskip\bigskip\bigskip
{\chota
This book is a significantly expanded version of author's earlier
introduction to \TeX, {\sl An Absolute Beginner's Guide to Using \TeX}
\thinspace{\copyr\ Arvind Borde, 1987). \smallskip
Apart from its covers, the book was typeset entirely in Plain \TeX\ by the
author, with assistance from the production department of Academic Press,
Boston. The covers were done partially in \TeX.
The final copy was produced by the American Mathematical Society on an
Autologic ASP Micro--5 phototypesetter. \smallskip
'\TeX' is a trademark of the American Mathematical Society.} \par}
\ejct

%***** DEDYKACJA *****
\vphantom{\title{}} w celu zapewnienia jednakowych odstępów stron tytułowych
\vskip 2,5 true in \rightline{\ded Moim rodzicom} \vfil\ejct
\bye

```

KONIEC

Nota tłumacza

Bogusław Jackowski

Dla mnie jest przyjemnością używanie narzędzia tak starannie przemyślanego jak \TeX , dającego naprawdę bogate możliwości. Poza tym, ja po prostu lubię dobrze złożone teksty.

Marek Ryćko

Ja nie mam obaw, że kiedykolwiek natknę się na coś, czego nie będę mógł złożyć \TeX -em. Ważne jest dla mnie również to, że stanowi on ustalony, szeroko akceptowalny standard, którego dalsza rozbudowa nie jest przewidziana. W związku z tym wiem, że warto uczyć się składać za jego pomocą, bo moja wiedza nie zestarzeje się za kilka lat. Tym bardziej, że działa on w sposób niezależny od sprzętu. Jeżeli pojawią się nowe komputery, systemy operacyjne, drukarki i monitory, to wiem, że \TeX tam będzie.

na podstawie proTypo, nr 2/1991

Pomysł tłumaczenia tej książki powstał kilka lat temu podczas konferencji Bacho \TeX '95*, jako reakcja na zarzuty o brak literatury \TeX -owej w rodzimym języku (książki, które pojawiają się na rynku dotyczą głównie La \TeX -a).

Niniejszy podręcznik umożliwia Czytelnikowi poznanie podstaw plain \TeX -a — w zakresie absolutnie minimalnym, ale zupełnie wystarczającym do tworzenia dobrej jakości składu tekstu. W stosunku do oryginału dokonano pewnych zmian, mających na celu przystosowanie do polskich zasad składu, a mianowicie:

- wykorzystano pakiet M \TeX (wspomagający skład \TeX -owy w języku polskim, autorstwa B. Jackowskiego i M. Ryćko), w miejsce oryginalnie opisywanych plików fontów `cm*` dokonano opisu i przytoczono w przykładach pliki fontów `pl*` — poprawnych typograficznie dla składu polskiego fontów rodziny Computer Modern;
- stosowane w Polsce znaki mniejsze-równe (\leq) i większe-równe (\geq) różnią się od znaków w fontach CM zaprojektowanych przez D. E. Knutha (odpowiednio: \leq i \geq) — uzyskano je za pomocą komend odpowiednio `\xleq` lub `\xle` oraz `\xgeq` lub `\xge`;
- to samo dotyczy polskiego nazewnictwa funkcji matematycznych „tg”, „ctg”, „tgh” i „ctgh” zamiast „tan”, „cot”, „tanh” i „coth” — zastosowano komendy `\tg`, `\ctg`,

* Bacho \TeX — od nazwy miejscowości Bachotek na pojezierzu Brodnickim; od roku 1993 polscy użytkownicy \TeX -a spotykają się tam na organizowanej przez Polską Grupę Użytkowników Systemu \TeX , corocznej konferencji (zobacz: <http://www.gust.org.pl>).

`\tgh`, `\ctgh` do składu nazw tych funkcji w formułach matematycznych; podobnie zapis funkcji „arc sin”, „arc cos” i tym podobne uzyskuje się konstrukcjami `\arc\sin`, `\arc\cos`;

- cyfry w tekście składane są zgodnie z polską normą, tj. separatorem miejsc dziesiętnych jest przecinek (T_EX także rozumie taki zapis, poprawnie interpretując zarówno zapis 10.5pt, jak też 10,5pt, natomiast sam wszystkie wymiary podaje z kropką, patrz np. opis komendy `\pagegoal` na str. 139);
- pierwotne plain-owe znaczenie komendy `\=` w polskim formacie M_EX, która użyta w miejsce łącznika (np. džentelmen-włamywacz, zapisywane `dżentelmen\=włamywacz`) zapewnia dzielenie słów składanych zgodnie z polską zasadą powtórzenia łącznika na początku drugiego wiersza; stosowny akcent uzyskuje się komendą `\macron` (np. `\macron a` składa `ā`);
- tekst przykładu ze stron 45–51 pozostawiono w wersji oryginalnej by pokazać możliwości składu w innych językach;
- w przykładach II, III, V, VII starano się zachować poprawność tłumaczenia z uwzględnieniem sposobu składu, przedkładając to ponad naukową wymowę tekstu — tu szczególnie podziękowania za merytoryczne sugestie dla Beaty Orchel;
- ze względu na styl publikacji dokonano kilkakrotnie minimalnej korekty formatu stron: raz — ze względu na charakter przykładów, dwa — na specyfikę tłumaczenia tekstów angielskich (teksty angielskie są na ogół bardziej zwarte); podobnie, stosując się do polskich zasad składu — nie pozostawiano spójników na końcach wierszy.

Do powstania tej książki w mniej lub bardziej znaczący sposób przyczyniło się wiele osób. Są to (w kolejności alfabetycznej): Włodek Bzyl, Maciej Głowacki, Bogusław Jackowski, Bogusław Lichoński, Krzysztof Leszczyński, Jurek Ludwichowski, Janusz M. Nowacki, Tomek Przechlewski, Marek Ryćko, Piotr Strzelczyk, Jola Szelatyńska, Andrzej Tomaszewski, Staszek Wawrykiewicz. . .

Szczególnie wdzięczni pozostajemy Beacie Orchel, za cenne wskazówki merytoryczne oraz Dorocie Zgaińskiej i Bogusławowi Jackowskiemu, za szczególnie mocną inspirację i wnikliwą korektę końcową — rezultatem ich pracy korektorsko-merytorycznej jest ostateczna wersja tłumaczenia.

Pomimo usilnych prób wydania drukiem niniejszego tłumaczenia — kilka lat leżało ono „na półce”. Zdając sobie sprawę że nie uda się tego materiału opublikować w żadnym wydawnictwie, udostępniamy go niniejszym wszystkim zainteresowanym Niech P.T. Czytelnicy będą od tej pory recenzentami naszej pracy. Za wszelkie uwagi będziemy wdzięczni, i nie omieszkamy ich uwzględnić oraz uaktualnić. **Jednocześnie zastrzegamy sobie, iż niniejszy materiał nie może być użytkowany komercyjnie, tak w całości jak i w fragmentach.**

Ewa Kmieciak, ek@pretext.com.pl
 Jacek Kmieciak, jk@pretext.com.pl

TeX w przykładach

Co należy wpisać:	Aby otrzymać:
<p>W książce tej wykorzystano nowoczesne podejście do nauki obsługi programów komputerowych: naukę za pomocą przykładów. Równolegle przedstawione są plik wejściowy składanego tekstu i wynik składu, co pozwala użytkownikowi szybko zorientować się, jak uzyskać np. symbol <code>\S</code>, jak złożyć tekst <code>{\it kursywą\}</code> czy krojem <code>{\bf pogrubionym}</code>, jak otrzymać wzór <code>\$\$ \beta^2 = \pi + \frac{a + 1}{2} , \$\$</code> jak rozpocząć nowy akapit, itd.</p> <p>Użytkownik może korzystać z podręcznika „wizualnie”, przeglądać strony aż do momentu, kiedy napotka interesującą go rzecz, lub od razu odszukać odpowiednie hasło w dodatku.</p> <p><code>{\sl Dodatek jest równocześnie indeksem i słownikiem --- zawiera krótkie wyjaśnienia każdego z haseł --- czyli doskonałym przewodnikiem po podstawowych zastosowaniach programu.}</code></p> <p><code>\centerline{\\$spadesuit\$}</code></p> <p>Unikając żargonu komputerowego, książka ta oferuje jasne i praktyczne wprowadzenie do TeX-a. Chociaż jest skierowana głównie do użytkowników „praktyków”, nie zaś do „dociekliwych” interesujących się wewnętrzną pracą TeX-a, mamy nadzieję, że wszyscy znajdą coś interesującego w przedstawionych w niej przykładach.</p> <p><code>\centerline{\\$star\$ \$star\$ \$star\$}</code></p>	<p>W książce tej wykorzystano nowoczesne podejście do nauki obsługi programów komputerowych: naukę za pomocą przykładów. Równolegle przedstawione są plik wejściowy składanego tekstu i wynik składu, co pozwala użytkownikowi szybko zorientować się, jak uzyskać np. symbol §, jak złożyć tekst <i>kursywą</i> czy krojem pogrubionym, jak otrzymać wzór</p> $\beta^2 = \pi + \frac{a + 1}{2},$ <p>jak rozpocząć nowy akapit, itd.</p> <p>Użytkownik może korzystać z podręcznika „wizualnie”, przeglądać strony aż do momentu, kiedy napotka interesującą go rzecz, lub od razu odszukać odpowiednie hasło w dodatku.</p> <p><i>Dodatek jest równocześnie indeksem i słownikiem — zawiera krótkie wyjaśnienia każdego z haseł — czyli doskonałym przewodnikiem po podstawowych zastosowaniach programu.</i></p> <p style="text-align: center;">♠</p> <p>Unikając żargonu komputerowego, książka ta oferuje jasne i praktyczne wprowadzenie do TeX-a. Chociaż jest skierowana głównie do użytkowników „praktyków”, nie zaś do „dociekliwych” interesujących się wewnętrzną pracą TeX-a, mamy nadzieję, że wszyscy znajdą coś interesującego w przedstawionych w niej przykładach.</p> <p style="text-align: center;">***</p>