

L<sup>A</sup>T<sub>E</sub>X

## T<sub>E</sub>X na goownicy\*

Hanna Kołodziejaska

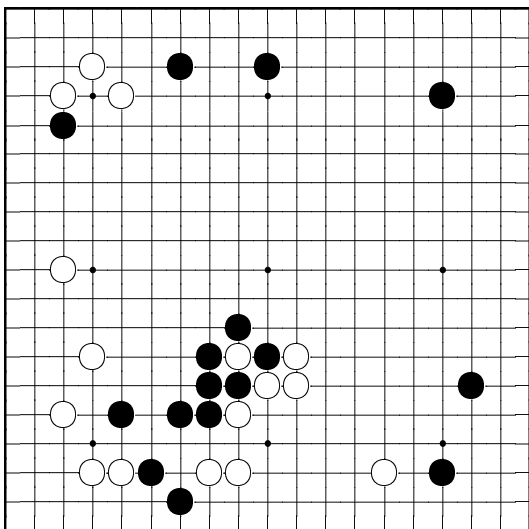
### 1 Wprowadzenie

Grą w Go zainteresowałam się kilka lat temu dzięki kolegom z Wydziału Matematyki, Informatyki i Mechaniki Uniwersytetu Warszawskiego. Usiłując zgłębić tajniki tej pasjonującej gry zaczęłam czytać literaturę goistyczną. Niewielki był pożytek z mojego czytania, gdyż bardziej niż strategia gry zaciekał mnie problem składania T<sub>E</sub>X-em diagramów. Ponadto w owym czasie trafiłam na artykuł Zalmana Rubinsteina o składaniu T<sub>E</sub>X-em diagramów szachowych [1]. I tak powstały moje fonty i makra do Go.

### 2 Co to jest Go

Go jest grą planszową, znaną od czterech tysięcy lat. Pochodzi z Chin, skąd poprzez Koreę i Japonię rozprzestrzeniło się na cały świat.

Planszę do Go stanowi drewniana deska z zaznaczonymi 19 poziomymi i 19 pionowymi liniami. Piony o kształcie soczewek nazywa się kamieniami, nawet gdy są wykonane z plastyku lub szkła. Podczas gry są one umieszczane na przecięciach linii (rys. 1).

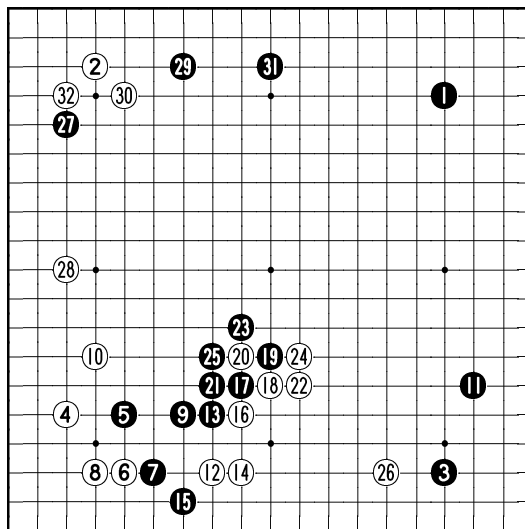


Rys. 1

\* Jest to oczywiście nazwa planszy do Go.

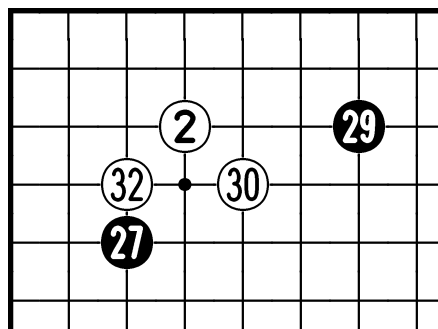
Na początku gry plansza jest pusta. Gracze na zmianę kładą na niej swoje pionki, po jednym w każdym ruchu. Jeden z graczy gra czarnymi pionkami, drugi białymi. Raz położony pion nie zmienia swojej pozycji (ewentualnie może być zdjęty z planszy w wyniku zbitcia).

Kolejność ruchów zaznacza się na diagramach numerując pionki: ❶ oznacza pierwszy pion w grze, ❷ — drugi itd. Przykładowa sytuacja na planszy z zaznaczoną kolejnością ruchów jest przedstawiona na rys. 2.



Rys. 2

Często zachodzi także potrzeba pokazania tylko pewnego fragmentu planszy. Może to być szczególna sytuacja podczas gry lub niezależny problem. Na rysunku 3 jest przedstawiony fragment planszy z rys. 2.

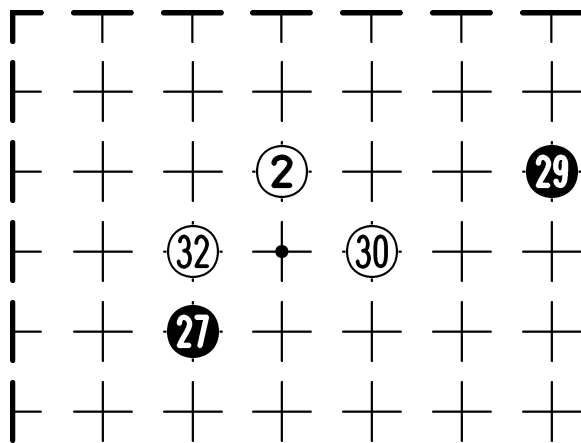


Rys. 3

### 3 Diagramy T<sub>E</sub>X-owe — definiowanie znaków w Metafoncie

Przystępując do pracy nad diagramami do Go zdecydowałam się zdefiniować i wygenerować przy pomocy METAFONTa wszystkie potrzebne symbole, nawet linie i kółka. W rezultacie sposób wstawiania diagramów do tekstu T<sub>E</sub>X-owego nie zależy od tego, czy pracujemy z plain T<sub>E</sub>X-em czy z też L<sup>A</sup>T<sub>E</sub>X-em.

Przyjrzyjmy się, z jakich symboli składa się diagram do Go. Weźmy, na przykład, diagram z rys. 3 i rozłożmy go na symbole składowe. Rezultat jest przedstawiony na rys. 4.



oraz  $\textcircled{2} = \textcircled{2} + \cdot$   
Rys. 4

Po przyjrzeniu się diagramowi rozłożonemu na proste elementy, możemy stwierdzić, że do składania diagramów T<sub>E</sub>X-em potrzebne są następujące znaki:

- białe pionki:  $\circ$ 
  - z numerami:  $\textcircled{1}$   $\textcircled{2}$   $\textcircled{3}$  itd.
  - z dodatkowymi symbolami:  $\triangle$   $\square$
- czarne pionki:  $\bullet$ 
  - z numerami:  $\textcircled{1}$   $\textcircled{2}$   $\textcircled{3}$  itd.
  - z dodatkowymi symbolami:  $\blacktriangle$   $\blacksquare$
- przecięcia linii:  $\text{+}$   $\text{+}$ 
  - linie boczne:  $\text{┃}$   $\text{┃}$  itd.
  - „uzupełnienia”:  $\cdot$   $\cdot$
  - „uzupełnienia” linii bocznych:  $\cdot$   $\cdot$  itd.

Skoncentrujmy się na pionkach białych. W przypadku czarnych rozumowanie oraz postępowanie jest identyczne. Przyjrzyjmy się następującym pionkom:



Zauważmy różnicę grubości i kształtu cyfr w kolejnych pionkach. Moje podejście było następujące:

1. Wygenerowałam znak  $\circ$  i zapamiętałam go pod nazwą `white_circle`.
2. Napisałam dziesięć METAFONT-owych makrodefinicji opisujących kształt cyfr: 1, 2, 3, 4, 5, 6, 7, 8, 9, 0, po czym zapamiętałam je również jako obiekty typu `picture`.
3. Za pomocą parametrów zmieniałam kształt cyfr oraz ich proporcje.

Oto moje makro do utworzenia znaku  $\circ$ :

```
beginchar(0,24/22size#,23/22size#,1/22size#);
pickup pencircle scaled line_thickness;
lft x1=1/24w; y1=11/24w;
rt x2=23/24w; y2=11/24w;
draw z1..z2..cycle;
white_circle := currentpicture;
showit; endchar;
```

Makro dla cyfry 1:

```
def digit_one =
  currentpicture := nullpicture;
  pickup pencircle scaled dig_pen;
  x1=.5dig_w; bot y1=0;
  x2=.5dig_w; top y2=dig_h;
  draw z1--z2;
  digit[1] := currentpicture;
  clearxy; clearit; clearpen;
enddef;
```

Fragment kodu w języku METAFONT dla znaku  $\textcircled{1}$ :

```
beginchar(1,24/22size#,23/22size#,1/22size#);
currentpicture := white_circle
+ digit[1] shifted
(.5w-.5dig_w,11/24w-.5dig_h);
showit;
endchar;
```

Spróbujmy teraz analogicznie otrzymać znak  $\textcircled{12}$ . Najpierw należy zmienić wartości parametrów określających szerokość cyfry i szerokość pióra (ang. *pen*) oraz wygenerować nowe obiekty typu `picture`.

```
dig_w := .7dig_w;
dig_pen := .7dig_pen;
digit_one; digit_two;
```

Znak  $\textcircled{12}$  możemy zdefiniować następująco:

```
beginchar(12,24/22size#,23/22size#,1/22size#);
currentpicture := white_circle
+ digit[1] shifted
(.5w-1.2dig_w,11/24w-.5dig_h)
+ digit[2] shifted
(.5w,11/24w-.5dig_h);
showit;
endchar;
```

Po wygenerowaniu wszystkich potrzebnych do Go znaków otrzymałam fonty trzech typów:

1. fonty zawierające czarne pionki, na przykład gobla10 (czarne pionki wielkości 10pt);
2. fonty zawierające białe pionki, na przykład gowhi10 (białe pionki wielkości 10pt);
3. fonty zawierające dodatkowe symbole (przecięcia linii, linie brzegowe itp.), np. go10 (wielkość 10pt).

Prawdopodobnie kolejne fonty z pionkami o numeracji powyżej 255 mogą okazać się potrzebne, ponieważ nie należą do rzadkości dłużej trwające gry!

#### 4 Makrodefinicje T<sub>E</sub>X-owe

Makrodefinicje potrzebne do kodowania diagramów do Go znajdują się w pliku go.sty. Jak wspomniałam, można ich używać zarówno w plain T<sub>E</sub>X-u jak i w L<sup>A</sup>T<sub>E</sub>X-u. Dołączenie pliku go.sty do składanego tekstu odbywa się komendą `\input` lub dodatkową opcją go L<sup>A</sup>T<sub>E</sub>X-owej komendy `\documentstyle`.

W literaturze do Go nie ma zwyczaju (o ile mi wiadomo) specjalnego oznaczania przecięć linii na planszy, czyli miejsc, na których są umieszczane pionki. Musiałam jednak wprowadzić oznaczenia, aby w wywołaniach makroinstrukcji T<sub>E</sub>X-owych dokładnie określić pozycję na planszy. W moich makrach każde przecięcie linii jest określone przez numer linii poziomej czyli *wiersza* (jedna z liter: a, b, c, d, e, f, g, h, i, k, l, m, n, o, p, q, r, s, t) i numer linii pionowej czyli *kolumny* (liczba od 1 do 19).

Zdefiniowałam następujące komendy T<sub>E</sub>X-owe wystarczające do składania diagramów do Go:

- wybór wielkości pionków i innych elementów diagramu (w punktach):

```
\gofontsize{10}
```

domyślnie: 10pt

- inicjalizacja całej planszy:

```
\inifullldiagram lub
```

```
\inidiagram_a-t:1-19_
```

(konieczna spacja ogranicza czwarty parametr)

- inicjalizacja fragmentu planszy:

```
\inidiagram_a-t:1-19_
```

(konieczna spacja ogranicza czwarty parametr) przy czym dwa pierwsze parametry określają wiersze, a dwa kolejne kolumny (od-do)

- określenie pionka:

```
\black.
```

```
\white.
```

```
\black{1}
```

```
\white{2}
```

```
\black{\square}
```

```
\white{\triangle}
```

przy czym kropka oznacza, że dany pion nie jest numerowany.

- inne określenie pozycji:

```
\letter{a}
```

```
\symbol{?}
```

- postawienie pionka (lub zaznaczenie ruchu):

```
\pos{a}{5}=\black{1}
```

```
\pos{a}{5}=\letter{a}
```

- pokazanie sytuacji na planszy:

```
\showfulldiagram lub
```

```
\showdiagram_a-t:1-19_
```

(konieczna spacja ogranicza czwarty parametr)

- pokazanie sytuacji na części planszy:

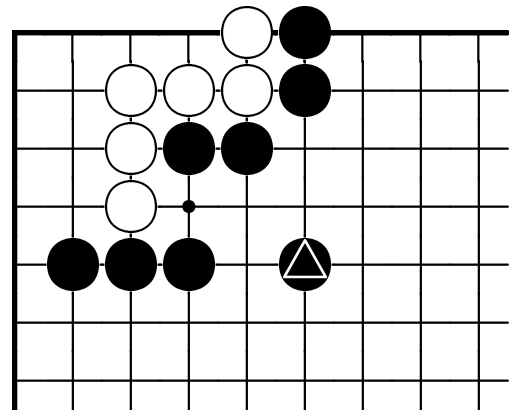
```
\showdiagram_a-t:1-19_
```

(konieczna spacja ogranicza czwarty parametr)

przy czym dwa pierwsze parametry określają wiersze, a dwa kolejne kolumny (od-do)

#### 5 Przykład

Na rysunku 5 został przedstawiony problem, a jego rozwiązanie na rys. 6.



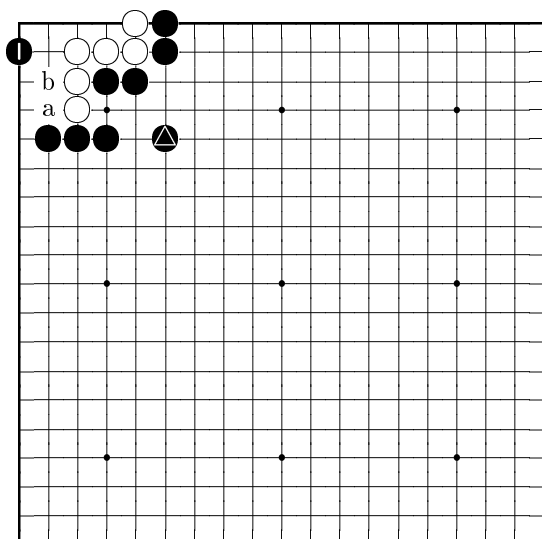
Rys. 5

```
\input go.sty %wczytanie pliku z makrami
\gofontsize{20} % wielkość pionków/symboli
% (domyślnie 10pt)
\pos{a}{5}=\white.
% położenie białego pionka na A5
% (bez żadnego numeru)
\pos{a}{6}=\black.
% położenie czarnego pionka na A6
% (bez żadnego numeru)
```

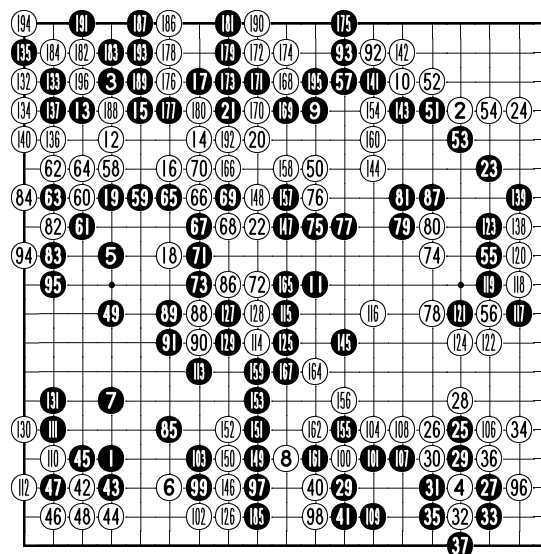
```

\pos{b}{3}=\white.
\pos{b}{4}=\white.
\pos{b}{5}=\white.
\pos{b}{6}=\black.
\pos{c}{3}=\white.
\pos{c}{4}=\black.
\pos{c}{5}=\black.
\pos{d}{3}=\white.
\pos{e}{2}=\black.
\pos{e}{3}=\black.
\pos{e}{4}=\black.
\pos{e}{6}=\black{\triangle}
% położenie czarnego piona
% oznaczonego trójkątem
$$
\showdiagram a-g:1-9
% wynik pokazany na rys. 5
$$
\pos{b}{1}=\black{1}
% położenie czarnego piona z 1
\pos{c}{2}=\letter{b}
% wpisanie litery ,b' na C2
\pos{d}{2}=\letter{a}
\gofontsize{10}
% zmiana wielkości pionów
% i innych symboli
$$
\showfulldiagram
% jak na rys. 6
$$
\inifulldiagram
% ponowna inicjalizacja planszy

```



Rys. 6



Rys. 7

Przykład prawdziwej gry przedstawia rys. 7.

## 6 T<sub>E</sub>X-owe makra od środka

Moim zasadniczym pomysłem było reprezentowanie każdego przecięcia linii na planszy przez T<sub>E</sub>X-ową komendę następującej postaci:

```
\@<litera><cyfra rzymska>
```

przy czym *litera* i *cyfra rzymska* określają odpowiednio wiersz i kolumnę. Oznacza to reprezentowanie planszy do Go jako tablicy 19×19 i pamiętanie wartości każdego jej elementu:

```

\@ai \@a1i \@a11i \@a1iv ... \@axix
\@bi \@b1i \@b11i \@b1iv ... \@bxix
.....
\@ti \@t1i \@t11i \@t1iv ... \@txix

```

Makrodefinicja `\pos`, o podstawowym znaczeniu dla składania diagramów, została zdefiniowana w następujący sposób:

```

\def\pos\#1\#2=\#3\#4{ .....
\expandafter\let
\csname@\#1\romannumerical\#2\endcsname=0
\relax
\edef\@pos{
\def\csname@\#1\romannumerical\#2
\endcsname{\#3{\@fourth}}\@pos
\ignorespaces}

```

Komenda `\edef` (*expanded definition*) jest używana także w innych makrach do składania diagramów do Go.

## 7 Uwagi końcowe

Diagramy są włączane do tekstu jako zwykłe pionowe pudełka (*vboxes*).

Piony mogą być także wstawiane bezpośrednio w tekst akapitu. Wówczas wystarczy użyć komend `\textwhite` i `\textblack` zamiast `\white` i `\black`. Na przykład zdanie z początku tego artykułu: „**1** oznacza pierwszy pion w grze, **2** — drugi” zostało złożone następująco: „`\textblack{1}` oznacza pierwszy pion w grze, `\textwhite{2}` — drugi”.

Nie ma żadnych innych tajemnic w składaniu  $\text{T}_{\text{E}}\text{X}$ -em diagramów do Go. Kształt cyfr użytych do numeracji pionów w grze jest daleki od doskonałości. Może ktoś podjąłby się utworzenia lepszych fontów?

Makrodefinicje  $\text{T}_{\text{E}}\text{X}$ -owe przeznaczone do składania diagramów  $9 \times 9$  lub  $13 \times 13$  można bez kłopotów dodać do pliku `go.sty` modyfikując istniejące makra dla diagramów  $19 \times 19$ .

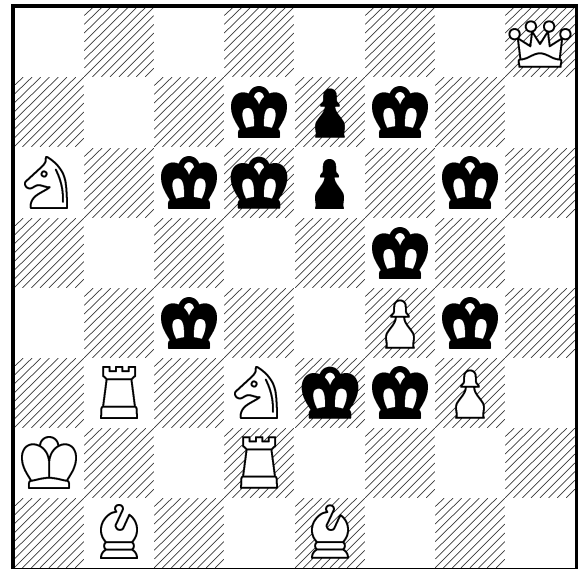
### Literatura

- [1] Zalman Rubinstein: *Chess printing via Metafont and T<sub>E</sub>X*. TUGboat Vol. 10, No. 2.
- [2] Iwamoto Kaoru, 9 dan: *Go for beginners*. Ishi Press, Inc., Tokyo 1972.
- [3] Janusz Kraszek: *Świat Go*. COK, Warszawa 1989.

## Jak nazywa się ten font?

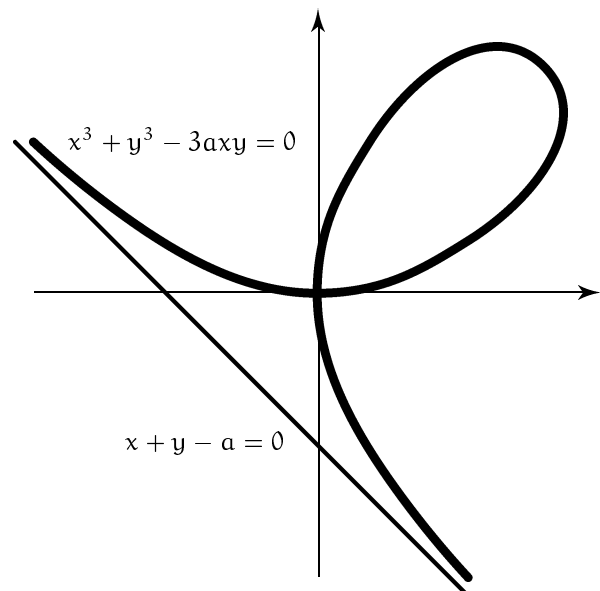


## Szachy



*Białe zaczynają i wygrywają*

## Wykres



## Uciecha z $\text{T}_{\text{E}}\text{X}$ -a

### Krzyżowka

W wolne pola wpisz komendy  $\text{T}_{\text{E}}\text{X}$ a.

\		G			
\		U			
\		S			
\		T			

**Plik go.sty**

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
%   This is 'go.sty' ver. 0.07 (September 1992)
%   (c) Copyright by Hanna Kołodziejska, 1992
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
\catcode'@=11

\newbox\@gobox \newdimen\@godimen

\def\gofontsize#1{
  \font\gofont=go#1 at #1truept
  \font\blackfont=gobla#1 at #1truept
  \font\whitefont=gowhi#1 at #1truept
  \ifnum #1=10 \font\letterfont=cmr10 at 10truept\else      %% <--- cm
    \ifnum #1=15 \font\letterfont=cmr10 at 14.4truept\else
      \ifnum #1=20 \font\letterfont=cmr10 at 17.28truept\fi\fi\fi
  \setbox\@gobox=\hbox{\gofont\char0}
  \@godimen=\wd\@gobox
}

\gofontsize{10} % initialization <---

\def\newgoline{\hfill\break}
\def\hoshi{\gofont\char0}
\def\empty{\gofont\char1}
\def\lftborder{\gofont\char2}
\def\rtborder{\gofont\char3}
\def\topborder{\gofont\char4}
\def\botborder{\gofont\char5}
\def\lfttopcorner{\gofont\char6}
\def\rttopcorner{\gofont\char7}
\def\lftbotcorner{\gofont\char8}
\def\rtbotcorner{\gofont\char9}
\def\triangle{\whitefont\char255}
\def\square{\whitefont\char254}

\newcount\n
\newcount\@beglet \newcount\@endlet
\newcount\@lettercount

\def\@letternumber#1{\relax
  \ifx #1a\@lettercount=1\else
  \ifx #1b\@lettercount=2\else
  \ifx #1c\@lettercount=3\else
  \ifx #1d\@lettercount=4\else
  \ifx #1e\@lettercount=5\else
  \ifx #1f\@lettercount=6\else
  \ifx #1g\@lettercount=7\else
  \ifx #1h\@lettercount=8\else
  \ifx #1i\@lettercount=9\else
  \ifx #1k\@lettercount=10\else
  \ifx #1l\@lettercount=11\else

```

```

\ifx #1m\@lettercount=12\else
\ifx #1n\@lettercount=13\else
\ifx #1o\@lettercount=14\else
\ifx #1p\@lettercount=15\else
\ifx #1q\@lettercount=16\else
\ifx #1r\@lettercount=17\else
\ifx #1s\@lettercount=18\else
\ifx #1t\@lettercount=19\else
  \errmessage{Row label must be letter!}
\fi\fi\fi\fi\fi\fi\fi\fi\fi\fi
\fi\fi\fi\fi\fi\fi\fi\fi\fi
}

\def\inidiagram#1-#2:#3-#4 {\relax
  \ifnum #3>#4 \errmessage{Invalid column numbers!} \fi
  \@letternumber{#1} \@beglet=\@lettercount
  \@letternumber{#2} \@endlet=\@lettercount
  \def\@inirow##1##2{\n=#3
    \loop
      \expandafter\let\csname @##1\romannumeral\n \endcsname=0\relax
      \edef\@inirow{\global
        \def\csname @##1\romannumeral\n \endcsname{##2}}\@inirow
      \ifnum \n<#4 \advance \n by 1
      \repeat
    }}\relax
  \loop
    \ifcase\@beglet\relax\or
    \@inirow{a}{\topborder}\relax
      \ifnum #3=1 \global\def\@ai{\lfttopcorner}\fi
      \ifnum #4=19 \global\def\@axix{\rttopcorner}\fi \or
    \@inirow{b}{\empty}\relax
      \ifnum #3=1 \global\def\@bi{\lftborder}\fi
      \ifnum #4=19 \global\def\@bxix{\rtborder}\fi \or
    \@inirow{c}{\empty}\relax
      \ifnum #3=1 \global\def\@ci{\lftborder}\fi
      \ifnum #4=19 \global\def\@cxix{\rtborder}\fi \or
    \@inirow{d}{\empty}\relax
      \ifnum #3=1 \global\def\@di{\lftborder}\fi
      \ifnum #3<5 \ifnum #4>3 \global\def\@div{\hoshi}\fi\fi
      \ifnum #3<11 \ifnum #4>9 \global\def\@dx{\hoshi}\fi\fi
      \ifnum #3<17 \ifnum #4>15 \global\def\@dxvi{\hoshi}\fi\fi
      \ifnum #4=19 \global\def\@dxix{\rtborder}\fi \or
    \@inirow{e}{\empty}
      \ifnum #3=1 \global\def\@ei{\lftborder}\fi
      \ifnum #4=19 \global\def\@exix{\rtborder}\fi \or
    \@inirow{f}{\empty}
      \ifnum #3=1 \global\def\@fi{\lftborder}\fi
      \ifnum #4=19 \global\def\@fxix{\rtborder}\fi \or
    \@inirow{g}{\empty}
      \ifnum #3=1 \global\def\@gi{\lftborder}\fi
      \ifnum #4=19 \global\def\@gxix{\rtborder}\fi \or
    \@inirow{h}{\empty}
      \ifnum #3=1 \global\def\@hi{\lftborder}\fi
      \ifnum #4=19 \global\def\@hxix{\rtborder}\fi \or
    \@inirow{i}{\empty}

```

```

\ifnum #3=1 \global\def\@ii{\lftborder}\fi
\ifnum #4=19 \global\def\@ixix{\rtborder}\fi \or
\@inirow{k}{\empty}
\ifnum #3=1 \global\def\@ki{\lftborder}\fi
\ifnum #3<5 \ifnum #4>3 \global\def\@kiv{\hoshi}\fi\fi
\ifnum #3<11 \ifnum #4>9 \global\def\@kx{\hoshi}\fi\fi
\ifnum #3<17 \ifnum #4>15 \global\def\@kxvi{\hoshi}\fi\fi
\ifnum #4=19 \global\def\@kxix{\rtborder}\fi \or
\@inirow{l}{\empty}
\ifnum #3=1 \global\def\@li{\lftborder}\fi
\ifnum #4=19 \global\def\@lxix{\rtborder}\fi \or
\@inirow{m}{\empty}
\ifnum #3=1 \global\def\@mi{\lftborder}\fi
\ifnum #4=19 \global\def\@mxix{\rtborder}\fi \or
\@inirow{n}{\empty}
\ifnum #3=1 \global\def\@ni{\lftborder}\fi
\ifnum #4=19 \global\def\@nxix{\rtborder}\fi \or
\@inirow{o}{\empty}
\ifnum #3=1 \global\def\@oi{\lftborder}\fi
\ifnum #4=19 \global\def\@oxix{\rtborder}\fi \or
\@inirow{p}{\empty}
\ifnum #3=1 \global\def\@pi{\lftborder}\fi
\ifnum #4=19 \global\def\@pxix{\rtborder}\fi \or
\@inirow{q}{\empty}
\ifnum #3=1 \global\def\@qi{\lftborder}\fi
\ifnum #3<5 \ifnum #4>3 \global\def\@qiv{\hoshi}\fi\fi
\ifnum #3<11 \ifnum #4>9 \global\def\@qx{\hoshi}\fi\fi
\ifnum #3<17 \ifnum #4>15 \global\def\@qxvi{\hoshi}\fi\fi
\ifnum #4=19 \global\def\@qxix{\rtborder}\fi \or
\@inirow{r}{\empty}
\ifnum #3=1 \global\def\@ri{\lftborder}\fi
\ifnum #4=19 \global\def\@rxix{\rtborder}\fi \or
\@inirow{s}{\empty}
\ifnum #3=1 \global\def\@si{\lftborder}\fi
\ifnum #4=19 \global\def\@sxix{\rtborder}\fi \or
\@inirow{t}{\botborder}
\ifnum #3=1 \global\def\@ti{\lftbotcorner}\fi
\ifnum #4=19 \global\def\@txix{\rtbotcorner}\fi
\fi
\ifnum \@beglet<\@endlet \advance \@beglet by 1
\repeat
}

```

```
\def\inifullldiagram{\inidiagram a-t:1-19 }
```

```
\inifullldiagram % initialization
```

```

\def\showdiagram#1-#2:#3-#4 {\vbox{\offinterlineskip\noindent
\ifnum #3>#4 \errmessage{Invalid column numbers!} \fi
\hsize=\@godimen
\n=#4\advance\n by-#3\advance\n by1
\multiply\hsize by\n
\@letternumber{#1} \@beglet=\@lettercount
\@letternumber{#2} \@endlet=\@lettercount
\def\@showrow##1{\n=#3

```



```

\loop
  \edef\@showrow{\csname @##1\romannumeral\n \endcsname}\@showrow
  \ifnum \n<#4 \advance \n by 1
  \repeat
  }}\relax
\loop
  \ifcase\@beglet\relax\or
  \@showrow{a}\or
  \@showrow{b}\or
  \@showrow{c}\or
  \@showrow{d}\or
  \@showrow{e}\or
  \@showrow{f}\or
  \@showrow{g}\or
  \@showrow{h}\or
  \@showrow{i}\or
  \@showrow{k}\or
  \@showrow{l}\or
  \@showrow{m}\or
  \@showrow{n}\or
  \@showrow{o}\or
  \@showrow{p}\or
  \@showrow{q}\or
  \@showrow{r}\or
  \@showrow{s}\or
  \@showrow{t}
  \fi
  \ifnum \@beglet<\@endlet
  \newgoline \advance \@beglet by 1
  \repeat
  }}

\def\showfulldiagram{\showdiagram a-t:1-19 }

\def\pos#1#2=#3#4{\relax
  \ifcat #1a\relax \else \errmessage{Row label must be a letter!}\fi
  \ifnum #2<1 \errmessage{Column number less than 1?}\fi
  \ifnum #2>19 \errmessage{Column number greater than 19?}\fi
  \edef\@fourth{#4}\relax
  \ifx .#4\ifx #3\empty\edef\@fourth{}\else \edef\@fourth{0}\fi\fi
  \edef\@borders{\relax
    \ifx #1a \ifnum #2=1 \gofont\char15 \else
      \ifnum #2=19 \gofont\char16 \else
        \gofont\char13 \fi\fi
    \else
      \ifx #1t \ifnum #2=1 \gofont\char17 \else
        \ifnum #2=19 \gofont\char18 \else
          \gofont\char14 \fi\fi
    \else
      \ifnum #2=1 \gofont\char11
      \else
      \ifnum #2=19 \gofont\char12
      \else
      \gofont\char10
      \fi\fi\fi\fi}\relax

```

```

\expandafter\let\csname @#1\romannumeral#2\endcsname=0\relax
\edef\@pos{\def\csname @#1\romannumeral#2\endcsname{#3{\@fourth}}}\@pos
\ignorespaces
}

\newcount\nr

\def\black#1{\relax
  \ifx \triangle#1{\let\whitefont=\blackfont
    \rlap{\triangle}\@borders}\else
  \ifx \square#1{\let\whitefont=\blackfont
    \rlap{\square}\@borders}\else
  \ifnum #1<0 \errmessage{Negative Black's move?}\else
  \ifnum #1>253 \errmessage{Black's move too big!}\else
  {\blackfont\rlap{\char#1}\@borders}\ignorespaces
  \fi\fi\fi\fi}

\def\white#1{\relax
  \ifx \triangle#1{\rlap{\triangle}\@borders}\else
  \ifx \square#1{\rlap{\square}\@borders}\else
  \ifnum #1<0 \errmessage{Negative White's move?}\else
  \ifnum #1>253 \errmessage{White's move too big!}\else
  {\whitefont\rlap{\char#1}\@borders}\ignorespaces
  \fi\fi\fi\fi}

\def\textblack#1{\def\@borders{\hskip\wd\@gobox}\leavevmode
  \ifx .#1\lower.2\wd\@gobox\hbox{\black{0}}\else
  \lower.2\wd\@gobox\hbox{\black{#1}}\fi}

\def\textwhite#1{\def\@borders{\hskip\wd\@gobox}\leavevmode
  \ifx .#1\lower.2\wd\@gobox\hbox{\white{0}}\else
  \lower.2\wd\@gobox\hbox{\white{#1}}\fi}

\newbox\@letterbox \newdimen\@letterdim

\def\symbol#1{\relax
  \ifcat a#1\relax\else
  \ifcat O#1\relax\else
    \errmessage{Strange parameter of symbol macro! (#1)}
  \fi\fi
  \setbox\@letterbox=\hbox to\wd\@gobox{\hfil\letterfont #1\hfil}\relax
  \@letterdim=\wd\@gobox
  \divide \@letterdim by 24
  \multiply \@letterdim by 22
  \advance \@letterdim by -\ht\@letterbox
  {\raise.5\@letterdim\box\@letterbox}\relax
}

\let\letter=\symbol

\catcode'\@=12
\endinput

```