METAFONT

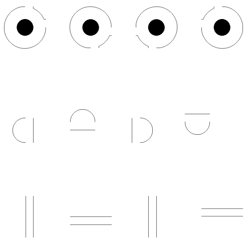
Odpowiedź na konkurs ogłoszony w biuletynie GUST-u nr 2, str. 27

Bogusław Lichoński

Przedstawienie ramki

Oto zgłoszona przeze mnie ramka:

Do wykreślenia powyższej ramki zaprojektowałem dwanaście następujących znaków:



Program METAFONT-owy rozpoczyna się (poza mode_setup) od nadania wartości parametrom:

- rw#, rh# określają szerokość pudełka w jakim umieszczę znaki fontu (w przykładzie pudełko ma wymiary 10×10 pt)
- q parametr określający odległość dwóch pionowych lub poziomych linii ramki (w przykładzie $q = 2$ pt)

```
rw#=10pt#; % szerokość pudełka
rh#=10pt#; % wysokość  pudełka
q=2pt;     % odległość między liniami
```

Definicję pierwszego znaku rozpoczynam od:

```
beginchar("3",rw#,rh#,0);
```

będzie to lewy górny narożnik ramki.

Deklaruję cztery użyteczne ścieżki, które posłużą mi do wyznaczania punktów kontrolnych.

```
path a; path b; path ml; path mll;
```

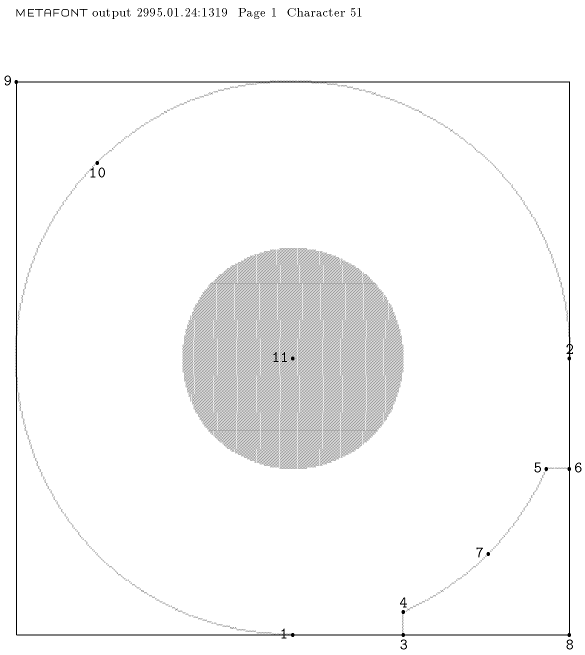
Kolejny krok to podanie współrzędnych (x, y) punktów z1, z2, z3, z6, z11

```
x1=x11=0.5w;
x3=x1+q;
x2=x6=2x1;
y2=y11=0.5h;
y6=y2-q;
y1=y3=0h;
```

oraz definicja czterech użytecznych ścieżek.

```
ml=(z1{right}..{up}z2..(0.5w,h));
mll=(z1{left}..(0,0.5h)..{down}z2);
a=((0.5w,0.5h)+(q,0)--z3);
b=(z6--(0,0.5h)-(0,q));
```

Przypatrzmy się teraz gotowemu już znakowi  (lewy górny narożnik ramki):



Zauważmy, że brakuje nam punktów z4, z5, z7, z8, z9, z10.

Pierwsze dwa nieznane jeszcze punkty wyznaczę jako przecięcie (dla punktu z4) ścieżki a ze ścieżką ml dla $q \geq 0$ lub a ze ścieżką mll dla $q < 0$;

```
if q>=0: z4=a intersectionpoint ml;
   else: z4=a intersectionpoint mll;
fi;
```

punkt z5 wyznaczam analogicznie

```
if q>=0: z5=b intersectionpoint ml;
   else: z5=b intersectionpoint mll;
fi;
```

Aby wyznaczyć punkt z7 definiuję dwa punkty pomocnicze (prawy-dolny i lewy-górny punkt pudełka) z8 i z9.

```
z8=(w,0);
z9=(0,h);
```

Punkt z7 (środek na małym łuku) wyznaczam jako przecięcie ścieżki `z8--z9` ze ścieżką `ml`.

```
z7=(z8--z9) intersectionpoint ml;
```

Kolejny punkt z10 (środek na dużym łuku) wyznaczam jako przecięcie ścieżki `z8--z9` ze ścieżką `z1{left}..{down}z2`

```
z10=(z8--z9) intersectionpoint
                    (z1{left}..{down}z2);
```

co kończy definicje punktów kontrolnych. Pozostało nam jedynie zdefiniować odpowiednie piórko:

```
pickup pencircle scaled 0.01pt;
```

i zaczynamy rysowanie:

```
draw z3--z4..z7..z5--z6;
draw z1{left}..z10..{down}z2;
```

Dorysujmy jeszcze kropę w środku, ale grubszym piórkiem.

```
pickup pencircle scaled 4pt;
drawdot z11;
```

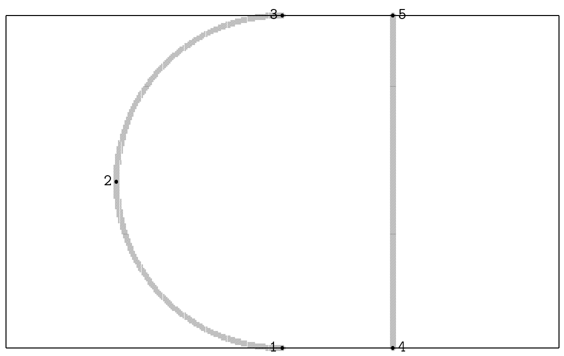
Na koniec zapisuję aktualnie utworzony obrazek pod zmienną LewyGorny i kończę definicję znaku.

```
picture LewyGorny;
   LewyGorny=currentpicture;
endchar;
```

Trzy pozostałe znaki narożnikowe ◉ ◉ ◉ definiuje się analogicznie. Są to symetrie osiowe powyższego znaku (patrz listing).

Kolejne cztery znaki ◖ ◠ ◗ ◡ są już dużo łatwiejsze do zdefiniowania. Weźmy jeden z nich:

METAFONT output 1995.01.25:1511 Page 1 Character 56



definiuję więc punkty

```
z1=(0.5w,0);
if q>0: z2=(0.2w,0.5h)
  else: z2=(0.2w+0.6w,0.5h)
fi;
z3=(0.5w,h);
z4=(0.5w+q,0);
z5=(0.5w+q,h);
```

i proszę METAFONT o narysowanie:

```
pickup pencircle scaled 0.1pt;
draw z1..z2..z3;
draw z4--z5;
```

Końcówka nie wymaga już wyjaśnień:

```
picture LewyPionowyBrzuszek;
LewyPionowyBrzuszek=currentpicture;
endchar;
```

wyjaśnienia wymaga jednak komenda warunkowa:

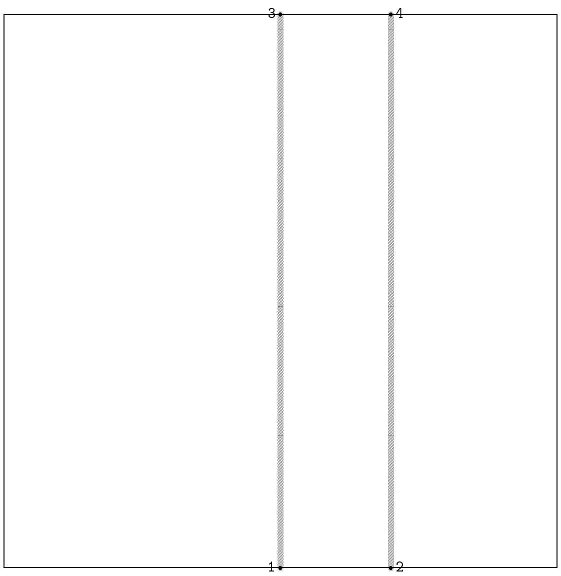
```
if q>0: z2=(0.2w,0.5h)
  else: z2=(0.2w+0.6w,0.5h)
fi;
```

stosuję tu taką instrukcję w celu „odwrócenia brzuszka” w przypadku, gdy q jest ujemne (patrz poniższy przykład).

Trzy pozostałe znaki „grzuszki” ◉ ◉ ◉ definiuje się analogicznie. Są to symetrie osiowe powyższego znaku lub obrót (patrz listing).

Pozostały jeszcze cztery najłatwiejsze do zdefiniowania znaki: ‖ ═ ‖ ═. Oto jeden z nich:

METAFONT output 1995.01.25:1513 Page 1 Character 49



i jego pełna definicja:

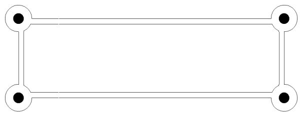
```
bot z1=(0.5w,0);
bot z2=(0.5w+q,0);
top z3=(0.5w,h);
top z4=(0.5w+q,h);
pickup pencircle scaled 0.1pt;
draw z1--z3;
draw z2--z4;
labels(1,2,3,4);
picture LewyPionowySlupek;
LewyPionowySlupek=currentpicture;
```

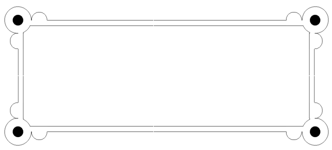
Przykłady

Z powyższych znaków można „złożyć” różne ramki. Pierwsza grupa ramek dla wartości:

```
q=2pt;
```

może wyglądać tak:

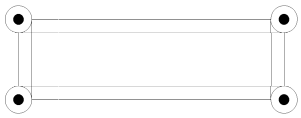


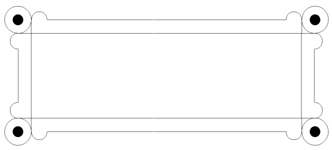


Wystarczy jednak zmienić parametr q tak oto:

```
q=5pt;
```

aby otrzymać:

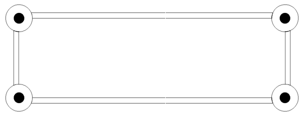


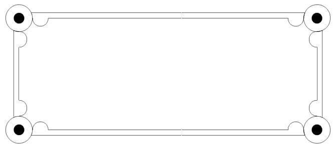


Zmieniając po raz kolejny parametr q na (q mniejsze od zera):

```
q=-2pt;
```

otrzymujemy†:





Mam nadzieję, iż moja ramka spodoba się Wysokiemu Jury.

Listing

```
%%%%% Plik ramka.mf wersja 2.6 %%%%%
mode_setup;
rw#=10pt#;
rh#=10pt#;
q=2pt;

beginchar("3",rw#,rh#,0);
path a; path b; path ml; path mll;
x1=x11=0.5w;
x3=x1+q;
x2=x6=2x1;
y2=y11=0.5h;
y6=y2-q;
y1=y3=0h;
ml=(z1{right}..{up}z2..(0.5w,h));
mll=(z1{left}..(0,0.5h)..{down}z2);
a=((0.5w,0.5h)+(q,0)--z3);
b=(z6--(0,0.5h)-(0,q));
if q>=0: z4=a intersectionpoint ml;
   else: z4=a intersectionpoint mll;
fi;
if q>=0: z5=b intersectionpoint ml;
   else: z5=b intersectionpoint mll;
fi;
z8=(w,0);
```

†: Zwróćmy uwagę na odwrócone brzuszki.

```
z9=(0,h);
z7=(z8--z9) intersectionpoint ml;
z10=(z8--z9) intersectionpoint
              (z1{left}..{down}z2);
pickup pencircle scaled 0.01pt;
draw z3--z4..z7..z5--z6;
draw z1{left}..z10..{down}z2;
pickup pencircle scaled 4pt;
drawdot z11;
labels(1,2,3,4,5,6,7,8,9,10,11);
picture LewyGorny;
         LewyGorny=currentpicture;
endchar;
%---------------------------------
beginchar("4",rw#,rh#,0);
currentpicture:=LewyGorny
reflectedabout((0.5w,0),(0.5w,h));
endchar;
%---------------------------------
beginchar("0",rw#,rh#,0);
currentpicture:=LewyGorny
reflectedabout((0,0.5h),(w,0.5h));
picture PrawyDolny;
        PrawyDolny=currentpicture;
endchar;
%---------------------------------
beginchar("5",rw#,rh#,0);
currentpicture:=PrawyDolny
reflectedabout((0.5w,0),(0.5w,h));
endchar;
%=================================
beginchar("8",rw#,0.6rh#,0);
z1=(0.5w,0);
if q>0: z2=(0.2w,0.5h)
  else: z2=(0.2w+0.6w,0.5h)
fi;
z3=(0.5w,h);
z4=(0.5w+q,0);
z5=(0.5w+q,h);
pickup pencircle scaled 0.1pt;
draw z1..z2..z3;
draw z4--z5;
labels(1,2,3,4,5);
picture LewyPionowyBrzuszek;
LewyPionowyBrzuszek=currentpicture;
endchar;
%---------------------------------
beginchar("9",rw#,0.6rh#,0);
currentpicture:=LewyPionowyBrzuszek
reflectedabout((0.5w,0),(0.5w,0.6h));
endchar;
%---------------------------------
beginchar("B",0.6rw#,rh#,0);
currentpicture:=LewyPionowyBrzuszek
rotatedabout((0.5w,0.3h),90);
picture DolnyPoziomyBrzuszek;
DolnyPoziomyBrzuszek=currentpicture;
endchar;
%---------------------------------
beginchar("A",0.6rw#,rh#,0);
currentpicture:=DolnyPoziomyBrzuszek
reflectedabout((0,0.5h),(w,0.5h));
endchar;
%=================================
beginchar("1",rw#,rh#,0);
bot z1=(0.5w,0);
bot z2=(0.5w+q,0);
top z3=(0.5w,h);
top z4=(0.5w+q,h);
pickup pencircle scaled 0.1pt;
draw z1--z3;
draw z2--z4;
labels(1,2,3,4);
picture LewyPionowySlupek;
LewyPionowySlupek=currentpicture;
endchar;
%---------------------------------
beginchar("7",rw#,rh#,0);
currentpicture:=LewyPionowySlupek
reflectedabout((0.5w,0),(0.5w,h));
picture PrawyPionowySlupek;
PrawyPionowySlupek=currentpicture;
endchar;
%---------------------------------
beginchar("6",rw#,rh#,0);
currentpicture:=PrawyPionowySlupek
rotatedabout((0.5w,0.5h),90);
picture GornyPoziomySlupek;
GornyPoziomySlupek=currentpicture;
endchar;
%---------------------------------
beginchar("2",rw#,rh#,0);
currentpicture:=GornyPoziomySlupek
reflectedabout((0,0.5h),(w,0.5h));
endchar;

end.
```

Ps. Od Redakcji

Zaprezentowany wyżej artykuł zwyciężył w naszym konkursie. Nagrodą jest książka N. Walsha: *Making TEX Work*, którą przekazujemy Bogusiowi Lichońskiemu. Gratulujemy! Listing programu METAFONT-owego można znaleźć w archiwum sunrise.pg.gda.pl: /TeX/GUST/bulletin/05.