

# Rysunki w T<sub>E</sub>X-u z metafontem i metapostem

Mariusz Czerniak

Uniwersyteckie Centrum Technologii Sieciowych  
Uniwersytet Mikołaja Kopernika w Toruniu

# O pakiecie

**Wersja:** 0.4.05

**Autorzy:** Thomas Leathrum, Geoffrey Tobin i Daniel Luecking

**Źródło:** <http://comp.uark.edu/~luecking/tex/mfpic.html>

## Instalacja

- `mfpic.tex` (`mfpic.sty`) – podstawowy plik pakietu,  
np. `/texmf/tex/generic/mfpic/`,
- `grafbase.mf` – makra metafonta,  
np. `/texmf/metafont/grafbase/`,
- `grafbase.mp`, `dvipsnam.mp` – makra metaposta i definicje kolorów,  
np. `/texmf/metapost/grafbase/`.

# Co dalej?

W dokumentach umieszczamy instrukcje:

plain T<sub>E</sub>X, pdfT<sub>E</sub>X, LaTeX209:

```
\input mfpic  
\usemetapost  
\opengraphicsfile{rys}  
:  
:  
\closegraphicsfile
```

LaTeX2e, pdfLaTeX:

```
\usepackage[metapost]{mfpic}  
\opengraphicsfile{rys}  
:  
:  
\closegraphicsfile
```

- 1) przetworzenie dokumentu, w wyniku którego powstaje plik \*.mp,
- 2) uruchomienie programu `mpost` z parametrem (nazwa pliku \*.mp), powstają wtedy rysunki w postaci plików eps, po jednym dla każdego otoczenia `mfpic` w dokumencie,
- 3) ponowne przetworzenie dokumentu,
- 4) otrzymanie pliku \*.ps z dołączonymi plikami \*.eps.

# Otoczenie mfpic

```
\mfpic[⟨skala x⟩][⟨skala y⟩]{⟨x0⟩}{⟨x1⟩}{⟨y0⟩}{⟨y1⟩}... \endmfpic
```

*skala x*, *skala y* – skalowanie `\mfpicunit`,

$x_0$ ,  $x_1$ ,  $y_0$ ,  $y_1$  – początek i koniec układu współrzędnych,

szerokość rysunku –  $(x_1 - x_0) \cdot (\textit{skala } x) \cdot \textit{\mfpicunit}$ ,

wysokość rysunku –  $(y_1 - y_0) \cdot (\textit{skala } y) \cdot \textit{\mfpicunit}$ ,

## Przykładowe polecenia

```
\axes, \xmarks{⟨x0⟩, ⟨x1⟩, ...}, \ymarks{⟨y0⟩, ⟨y1⟩, ...},
```

```
\lines{⟨p0⟩, ⟨p1⟩, ...}, \polygon{⟨p0⟩, ⟨p1⟩, ...}, \rect{⟨p0⟩, ⟨p1⟩},
```

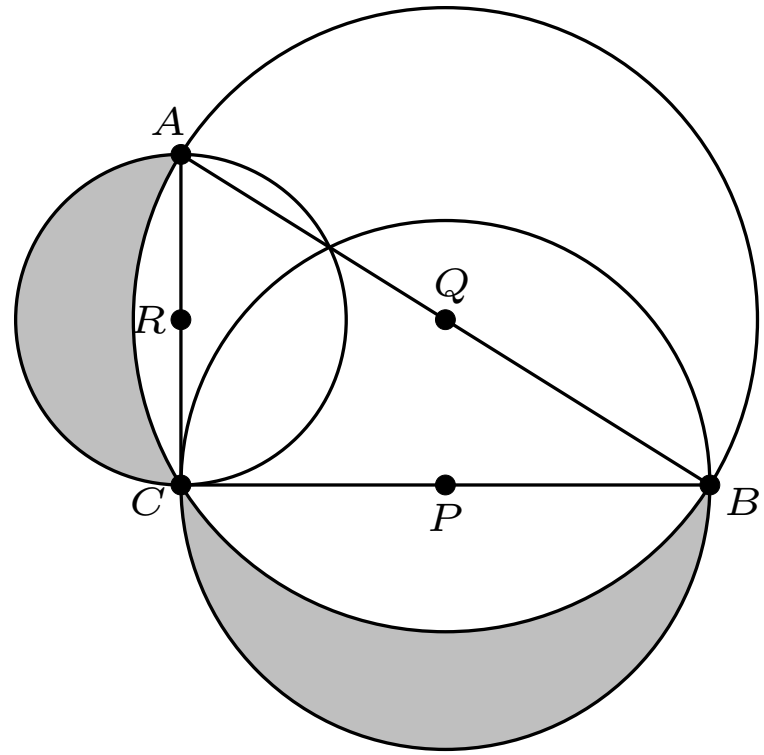
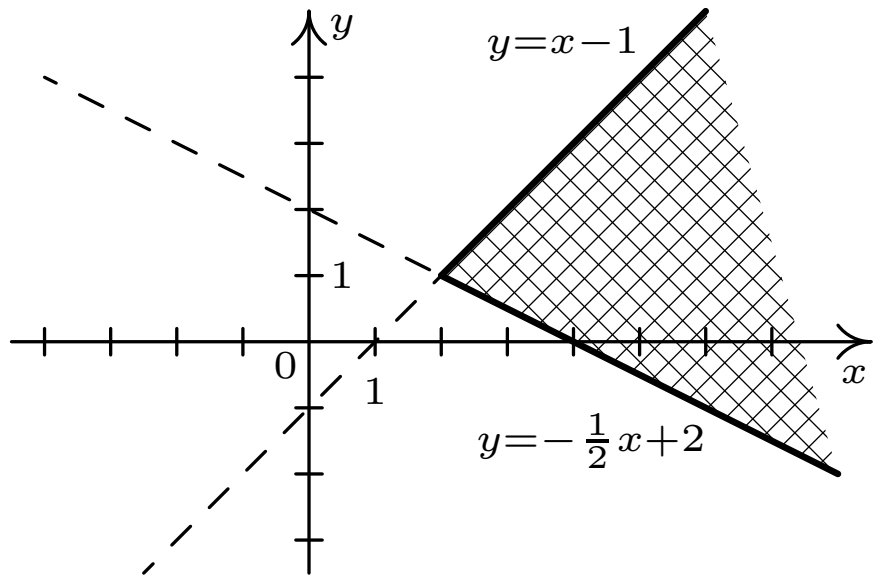
```
\circle{⟨c⟩, ⟨r⟩}, \ellipse[⟨θ⟩]{⟨c⟩, ⟨a⟩, ⟨b⟩}, \curve{⟨p0⟩, ⟨p1⟩, ...},
```

```
\arrow..., \dotted..., \dashed..., \gfill..., \gclear..., \shade...,
```

```
\thatch[⟨odległość⟩, ⟨kąt⟩], \lhatch[⟨odległość⟩], \rhatch..., \xhatch...,
```

```
\function{⟨xmin⟩, ⟨xmax⟩, ⟨krok⟩}{⟨f⟩},
```

$+$ ,  $-$ ,  $*$ ,  $/$ ,  $**$ ,  $()$ , `sqrt`, `sind`, `cosd`, `mlog(x) = 256 ln(x)`, `mexp`



# Kolorowe rysunki

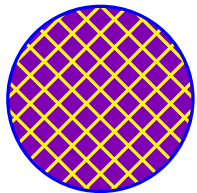
- `\drawcolor{<kolor>}`
- `\fillcolor{<kolor>}`
- `\hatchcolor{<kolor>}`
- `\headcolor{<kolor>}`

*kolor* może być przedstawiony jako:

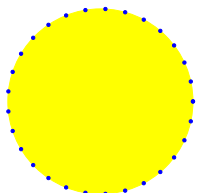
- black, white, green, red, blue, cyan, magenta, yellow,
  - wyrażenie, np. `red + green`, `0.5*red + 0.5*yellow`, `0.35*red`
  - układ wartości modelu RGB – `(1,0,0)`, `rgb(1,0,0)`, `RGB(255,0,0)`,
  - układ wartości modelu CMYK – `cmyk(1,1,0,0)`,
  - szarości: `0.7*white`, `gray(.7)`.
- `\mfpdefinicolor{<nazwa>}{<model>}{<specyfikacja>}`,
- przykład: `\mfpdefinicolor{DarkPuce}{rgb}{.8,.12,.56}`,
- użycie: `\drawcolor[named]{DarkPuce}`.

# Dodatkowa opcja koloru

- `\gfill[⟨kolor⟩]...`
- `\draw[⟨kolor⟩]...`
- `\point[⟨rozmiar⟩][⟨kolor⟩]...`
- `\thatch[⟨odległość⟩,⟨kąt⟩][⟨kolor⟩]...`
- `\hatch[⟨odległość⟩][⟨kolor⟩]...`
- `\arrow[l⟨długość⟩][r⟨kąt⟩][b⟨przesunięcie⟩][⟨kolor⟩]...`



```
\hatchcolor{yellow}
\draw[blue]\hatch\gfill[0.5*red + 0.7*blue]
\circle{(0,0),2}
```



```
\drawcolor{blue}
\dotted\gfill[yellow+red]\circle{(0,0),2}
\drawcolor{black}
```

## Rekordowe wyniki biegów człowieka i konia

