

# METAPOST

Jethro

PŠMF 2010

- makrojazyk pro tvorbu obrázků
- vektorový výstup v PS či PDF
- T<sub>E</sub>Xovské popisky
- řešení soustav lineárních rovnic

- METAPOST je součástí balíku T<sub>E</sub>Xlive
- výstup do Postscriptu pomocí `mpost`
- výstup do PDF pomocí `mptopdf`
- obrázky pojmenovávány jako
  - `název_zdrojáku.číslo_obrázku` v PS
  - `název_zdrojáku-číslo_obrázku.pdf` v PDF

- numeric – čísla
- pair – souřadnice
- path – křivka
- transform – afinní transformace
- color – barva
- string – řetězec
- boolean – pravdivostní proměnná
- picture – obrázek
- pen – pero

- reprezentovány  $k$ -násobkem zlomku  $\frac{1}{65536}$
- absolutní hodnota nejvýše 4096
- výsledky až  $8\times$  větší

# Deklarace proměnných

- proměnná se deklaruje `typ název_proměnné`
- pole se deklaruje `typ název_proměnné[]`
- nedeklarované proměnné považované za `numeric`
- proměnné začínající `z` považované za `pair`
- pokud je v názvu proměnné tečka, je ignorována  
např. `m.a == ma`

```
prologues := 1 ;  
...  
% spolecne definice  
...  
beginfig ( 1 ) ;  
...  
% kod prvnioho obrazku  
...  
endfig ;  
beginfig ( 42 ) ;  
...  
% kod druheho obrazku  
...  
endfig ( ) ;
```

# A pojďme kreslit

```
pair A;  
pair B;  
pair C;  
pair D;  
u=1cm;  
A = (0,0);  
B = (0,1u);  
C = (1u,1u);  
D = (1u,0);  
  
draw A -- B -- C -- D -- cycle;
```



# Transformace

- shifted *vektor* – posunutí
- rotated *uhel* – rotace okolo počátku
- rotatedaround (*vektor*, *uhel*) – rotace okolo daného bodu
- slanted *cislo* – zkosení
- scaled *koeficient* – zvětšení či zmenšení
- xscaled *koeficient* – zvětšení či zmenšení x-ové části
- yscaled *koeficient* – zvětšení či zmenšení y-ové části

```
A = (0,0);
```

```
B = (0,1u);
```

```
C = (1u,1u);
```

```
D = B rotated 90;
```

```
draw A — B — C — D — cycle ;
```

# For cyklus

- METAPOST je makrojazyk, tudíž lze psát kdekoli
- **for** i=1 **upto** 10: ... **endfor**
- **for** i=1 **step** 0.5 **until** 5: ... **endfor**
- **forever**: ... **exitif** bool **endfor**
- **forever**: ... **exitunless** bool **endfor**

```
A := (1u,0) rotated 45;
```

```
draw A — — for i=0 upto 3:
```

```
  A rotated (90*(i)) — — endfor cycle;
```

- každá křivka je cesta
- každá cesta má čas
- časy jednoznačně určují body na cestě
- využití u průsečíků, podmnožinách cest, ....

## Příklady

- `length`  $a$  – čas cesty
- `point`  $t$  of  $cesta$  – souřadnice bodu o čase  $t$
- `subpath`  $(t_1, t_2)$  of  $cesta$  – podcesta mezi časy  $t_1$  a  $t_2$
- `a` `intesectiontimes`  $b$  – vrací dvojici časů průsečíku na obou křivkách

- `label.pozice (text, bod)`
  - *pozice* určuje relativní umístění vůči *bod*  
`lft, rt, top, bot, llft, lrt, ulft, urt`
  - *text* je text popisku, pokud chceme T<sub>E</sub>X, použijeme  
`btex text etex`
  - *bod* určuje umístění popisku
- pokud chceme v místě popisku udělat tečku, použijeme  
`dotlabel`
- T<sub>E</sub>Xové popisky nelze použít (bez triku) ve for cyklu

- Miroslava Krátká: Tvorba obrázků pro matematické texty pomocí METAPOSTu  
<http://tex.loria.fr/prod-graph/kratka-diplomka2001.pdf>
- Metapost:  
<http://www.tug.org/docs/html/metapost/>
- André Heck: Learning METAPOST by Doing  
[http://www.tlhiv.org/MetaPost/documentation/MetaPost\\_Learning.pdf](http://www.tlhiv.org/MetaPost/documentation/MetaPost_Learning.pdf)
- Ukázky:  
<http://tex.loria.fr/prod-graph/zoonekynd/metapost/metapost.html>