


Egy jó kép felér ezer szóval.

Képek és rajzok elősegítik az előadás tartalmának jobb megértését.

 egy dián egy kép

Felhasznált anyagok:

- Jan-Philipp Kappmeier How to TikZ?
- Meik Hellmund PGF/TikZ- Graphics for \LaTeX
- Norbert Manthey A brief introduction into TikZ

Képek szerkesztése.

- pixel grafika
 - saját fényképek
 - Photoshop
 - Paint
 - stb
- vektor grafika
 - Inkscape
 - CorelDraw
 - Adobe Illustrator
 - TikZEd
 - stb.

TikZ

- TikZ vektor grafikát leíró nyelv
- Két szintet különböztetünk meg
 - TikZ könnyen használható \LaTeX parancsok gyűjteménye
 - PGF (Portable Graphics Format) alacsony szintű parancsok.

Előnyök:

- Illeszkedik a dokumentumhoz.
- Tetszőleges \LaTeX parancsokat használhatunk.
- Beamer teljes mértékben támogatja (animációk).
- Néhány rajzprogram a képet ebbe a formába is ki tudja menteni.

Hátrányok:

- No WYSIWYG
- Egy ábra elkészítése munkás

Kezdet

- 1 TikZ csomag beillesztése
- 2 Egyéb TikZ könyvtárak megadása (ha szükséges)
- 3 TikZ kód megírása:
 - használjuk a `tikzpicture` környezetet
 - inline TikZ a `\tikz` paranccsal

Példa

```
\usepackage{tikz}
\usetikzlibrary{shapes,calc}
...
\begin{tikzpicture}
...
\end{tikzpicture}
...
\tikz \filldraw[color=Thistle] circle (0.5ex); ●
```

TikZ alapok

Általános elvek

- A fő szintaktikai elem a *path* (nyomvonal).
- A *path* a koordináták és parancsok sorozata
- Általános szintaktika:
`\draw`[options] (koordinára) parancs (koordináta) ...
Értelmezés: menj a ceruzával a megadott helyre, rajzolj valamit ...
- Különböző parancsok lehetnek
 - `\draw` – vonal és alakzat rajzolás
 - `\fill` – belső terület kitöltése
 - `\node` – szöveget tartalmazó zárt alakzat
 - `\coordinate` – láthatatlan koordináta pont megadás

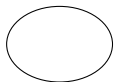
Alakzatok



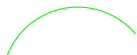
```
\draw (0, 0) rectangle (2, 1);
```



```
\draw[color=red] (0, 0) circle (.5);
```



```
\draw (0, 0) ellipse (.7 and 0.5);
```



```
\draw[color=green] (40:1) arc (40:160:1);
```

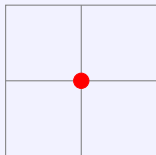


```
\draw[line width=2pt] (0, 0) ..  
    controls(1,1) .. (3, 0);
```

Path rajzolás

- `\draw` parancs lépésről lépésre.

Parancsok



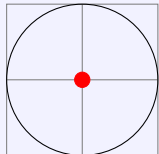
```
\begin{tikzpicture}
\path[draw] (0,0) circle (0.5)
            (-0.5,-0.5) -- (0.5,0.5) ;
\end{tikzpicture}
```

- Menjünk a $(0,0)$ koordináta pontba.
- Ne rajzoljunk semmit.

Path rajzolás

- `\draw` parancs lépésről lépésre.

Parancsok



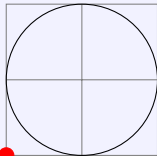
```
\begin{tikzpicture}
\path[draw] (0,0) circle (0.5)
            (-0.5,-0.5) -- (0.5,0.5) ;
\end{tikzpicture}
```

- Rajzoljunk egy 0.5 sugarú kört.
- Az aktuális koordináta értéke nem változik.

Path rajzolás

- `\draw` parancs lépésről lépésre.

Parancsok



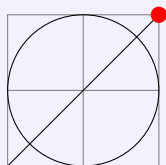
```
\begin{tikzpicture}
\path[draw] (0,0) circle (0.5)
            (-0.5,-0.5) -- (0.5,0.5) ;
\end{tikzpicture}
```

- Menjünk a $(-0.5,-0.5)$ koordináta pontba.

Path rajzolás

- `\draw` parancs lépésről lépésre.

Parancsok



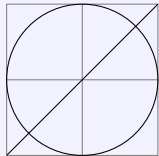
```
\begin{tikzpicture}
\path[draw] (0,0) circle (0.5)
            (-0.5,-0.5) -- (0.5,0.5) ;
\end{tikzpicture}
```

- Rajzoljunk szakaszt a $(0.5,0.5)$ pontig.

Path rajzolás

- `\draw` parancs lépésről lépésre.

Parancsok




```
\begin{tikzpicture}  
\path[draw] (0,0) circle (0.5)  
            (-0.5,-0.5) -- (0.5,0.5) ;  
\end{tikzpicture}
```

- Path parancs vége.

Második példa

- `\filldraw` parancs lépésről lépésre.

Geometriai formák



```
\filldraw[fill=Periwinkle,thick] (1,0)
rectangle +(2,1) -- (3,2);
```

- Menjünk a $(1,0)$ koordináta pontba.
- Nem rajzoljunk semmit.

Második példa

- `\filldraw` parancs lépésről lépésre.

Geometriai formák



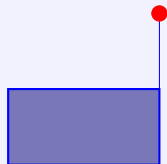
```
\filldraw[fill=Periwinkle,thick] (1,0)  
rectangle +(2,1) -- (3,2);
```

- Rajzoljunk egy 2 széles 1 magas négyzetet.
- Az aktuális koordináta értéke változik.

Második példa

- `\filldraw` parancs lépésről lépésre.

Geometriai formák



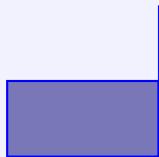
```
\filldraw[fill=Periwinkle,thick] (1,0)  
rectangle +(2,1) -- (3,2);
```

- Húzzunk vonalat a $(3,2)$ pontba.
- Az aktuális koordináta értéke változik.

Második példa

- `\filldraw` parancs lépésről lépésre.

Geometriai formák



```
\filldraw[fill=Periwinkle,thick] (1,0)  
rectangle +(2,1) -- (3,2);
```

- Path parancs vége.

Harmadik példa

- Vonal rajzoló parancs lépésről lépésre.

Kis ház rajzolás



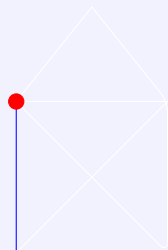
```
\draw<9> (0,0) -- (0,2) -- (1,3.25)
          {[rounded corners] -- (2,2) -- (2,0)}
          -- (0,2) -- (2,2) -- (0,0) -- (2,0);
```

- Menjünk a $(0,0)$ koordináta pontba.
- Nem rajzoljunk semmit.

Harmadik példa

- Vonalt rajzoló parancs lépésről lépésre.

Kis ház rajzolás



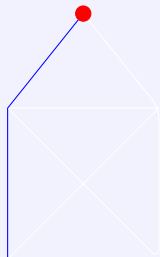
```
\draw<9> (0,0) -- (0,2) -- (1,3.25)
  {[rounded corners] -- (2,2) -- (2,0)}
  -- (0,2) -- (2,2) -- (0,0) -- (2,0);
```

- Húzzunk vonalat a $(0,2)$ pontba.
- Az aktuális koordináta értéke változik.

Harmadik példa

- Vonalt rajzoló parancs lépésről lépésre.

Kis ház rajzolás



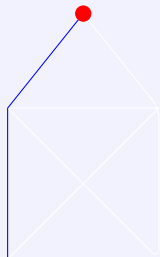
```
\draw<9> (0,0) -- (0,2) -- (1,3.25)
  {[rounded corners] -- (2,2) -- (2,0)}
  -- (0,2) -- (2,2) -- (0,0) -- (2,0);
```

- Húzzunk vonalat a $(1,3.25)$ pontba.
- Az aktuális koordináta értéke változik.

Harmadik példa

- Vonal rajzoló parancs lépésről lépésre.

Kis ház rajzolás



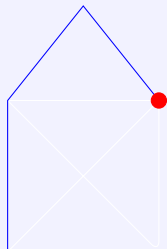
```
\draw<9> (0,0) -- (0,2) -- (1,3.25)
  {[rounded corners] -- (2,2) -- (2,0)}
  -- (0,2) -- (2,2) -- (0,0) -- (2,0);
```

- Lekerekített sarkok definiálása.
- Nem rajzoljunk semmit.

Harmadik példa

- Vonalt rajzoló parancs lépésről lépésre.

Kis ház rajzolás



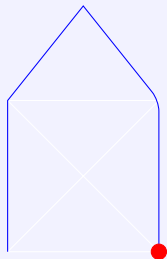
```
\draw<9> (0,0) -- (0,2) -- (1,3.25)
  {[rounded corners] -- (2,2) -- (2,0)}
  -- (0,2) -- (2,2) -- (0,0) -- (2,0);
```

- Húzzunk vonalat a **(2,2)** pontba.
- Az aktuális koordináta értéke változik.

Harmadik példa

- Vonalt rajzoló parancs lépésről lépésre.

Kis ház rajzolás



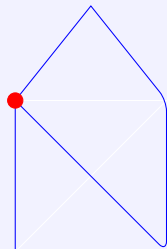
```
\draw<9> (0,0) -- (0,2) -- (1,3.25)
  {[rounded corners] -- (2,2) -- (2,0)}
  -- (0,2) -- (2,2) -- (0,0) -- (2,0);
```

- Húzzunk vonalat a $(2,0)$ pontba.
- Az aktuális koordináta értéke változik.

Harmadik példa

- Vonalt rajzoló parancs lépésről lépésre.

Kis ház rajzolás



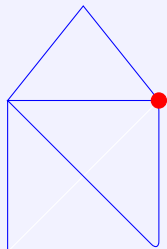
```
\draw<9> (0,0) -- (0,2) -- (1,3.25)
  {[rounded corners] -- (2,2) -- (2,0)}
  -- (0,2) -- (2,2) -- (0,0) -- (2,0);
```

- Húzzunk vonalat a **(0,2)** pontba.
- Az aktuális koordináta értéke változik.

Harmadik példa

- Vonalt rajzoló parancs lépésről lépésre.

Kis ház rajzolás



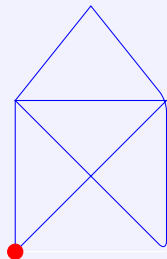
```
\draw<9> (0,0) -- (0,2) -- (1,3.25)
  {[rounded corners] -- (2,2) -- (2,0)}
  -- (0,2) -- (2,2) -- (0,0) -- (2,0);
```

- Húzzunk vonalat a **(2,2)** pontba.
- Az aktuális koordináta értéke változik.

Harmadik példa

- Vonalt rajzoló parancs lépésről lépésre.

Kis ház rajzolás



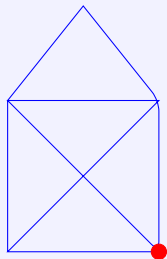
```
\draw<9> (0,0) -- (0,2) -- (1,3.25)
  {[rounded corners] -- (2,2) -- (2,0)}
  -- (0,2) -- (2,2) -- (0,0) -- (2,0);
```

- Húzzunk vonalat a $(0,0)$ pontba.
- Az aktuális koordináta értéke változik.

Harmadik példa

- Vonalt rajzoló parancs lépésről lépésre.

Kis ház rajzolás



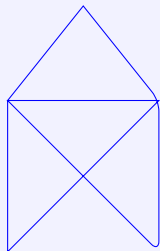
```
\draw<9> (0,0) -- (0,2) -- (1,3.25)
  {[rounded corners] -- (2,2) -- (2,0)}
  -- (0,2) -- (2,2) -- (0,0) -- (2,0);
```

- Húzzunk vonalat a $(2,0)$ pontba.
- Az aktuális koordináta értéke változik.

Harmadik példa

- Vonalt rajzoló parancs lépésről lépésre.

Kis ház rajzolás

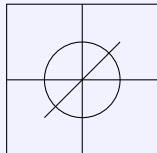


```
\draw<9> (0,0) -- (0,2) -- (1,3.25)
  {[rounded corners] -- (2,2) -- (2,0)}
  -- (0,2) -- (2,2) -- (0,0) -- (2,0);
```

- Path parancs vége.

TikZ opciók

Példa

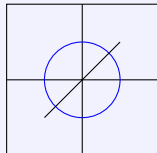


```
\begin{tikzpicture}
\draw (0,0) circle (0.5);
\draw (-0.5,-0.5) -- (0.5,0.5);
\end{tikzpicture}
```

- Rajz az alapértelmezett opciókkal.

TikZ opciók

Példa

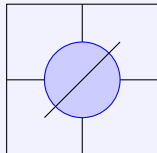


```
\begin{tikzpicture}
\draw [color=blue ]
      (0,0) circle (0.5);
\draw (-0.5,-0.5) -- (0.5,0.5);
\end{tikzpicture}
```

- Kör színének a megváltoztatása

TikZ opciók

Példa



```

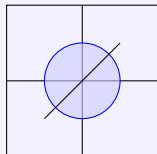
\begin{tikzpicture}
\draw [color=blue ,
      fill=blue!20white]
      (0,0) circle (0.5);
\draw (-0.5,-0.5) -- (0.5,0.5);
\end{tikzpicture}

```

- Kör belsejének a kitöltése.

TikZ opciók

Példa



```

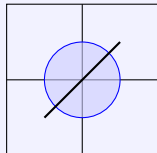
\begin{tikzpicture}
\draw [color=blue ,
      fill=blue!20white, fill opacity=0.6]
      (0,0) circle (0.5);
\draw (-0.5,-0.5) -- (0.5,0.5);
\end{tikzpicture}

```

- Átlátszóság definiálás.

TikZ opciók

Példa



```

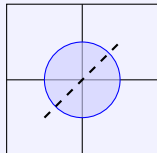
\begin{tikzpicture}
\draw [color=blue ,
      fill=blue!20white]
      (0,0) circle (0.5);
\draw[thick] (-0.5,-0.5) -- (0.5,0.5);
\end{tikzpicture}

```

- Vonal megvastagítása.

TikZ opciók

Példa



```

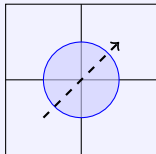
\begin{tikzpicture}
\draw [color=blue ,
      fill=blue!20white]
      (0,0) circle (0.5);
\draw[thick,dashed] (-0.5,-0.5) -- (0.5,0.5);
\end{tikzpicture}

```

- Szaggatott vonal rajzolás.

TikZ opciók

Példa



```

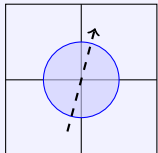
\begin{tikzpicture}
\draw [color=blue ,
      fill=blue!20white]
      (0,0) circle (0.5);
\draw[thick,dashed,->]
      (-0.5,-0.5) -- (0.5,0.5);
\end{tikzpicture}

```

- Nyíl rajzolása.

TikZ opciók

Példa



```

\begin{tikzpicture}
\draw [color=blue ,
      fill=blue!20white]
      (0,0) circle (0.5);
\draw[thick,dashed,->,rotate=30]
      (-0.5,-0.5) -- (0.5,0.5);
\end{tikzpicture}

```

- Forgatás 30° -kal.

Egyszerre több objektum opcióinak a beállítása

- Opciók beállításának lehetőségei.

Példa



```
\begin{tikzpicture}
\draw[orange,very thick] (0,0) circle (0.5);
\draw[orange,very thick] (-0.5,-0.5)--(0.5,0.5);
\draw[orange,very thick] (0.5,-0.5)--(-0.5,0.5);
\end{tikzpicture}
```

- Minden objektumra külön adjuk meg az opciókat.

Egyszerre több objektum opcióinak a beállítása

- Opciók beállításának lehetőségei.

Példa



```
\begin{tikzpicture}[orange,very thick]
\draw(0,0) circle (0.5);
\draw(-0.5,-0.5)--(0.5,0.5);
\draw(0.5,-0.5)--(-0.5,0.5);
\end{tikzpicture}
```

- Opciók megadása a teljes rajzra.

Egyszerre több objektum opcióinak a beállítása

- Opciók beállításának lehetőségei.

Példa



```

\begin{tikzpicture}
\draw(0,0) circle (0.5);
\begin{scope}[orange, very thick]
\draw(-0.5,-0.5)--(0.5,0.5);
\draw(0.5,-0.5)--(-0.5,0.5);
\end{scope}
\end{tikzpicture}

```

- Opciók megadása szelektíven.

Egyszerre több objektum opcióinak a beállítása

- Opciók beállításának lehetőségei.

Példa



```
\begin{tikzpicture}
    [marked/.style={orange,very thick}]
\draw(0,0) circle (0.5);
\draw[marked] (-0.5,-0.5)--(0.5,0.5);
\draw[marked] (0.5,-0.5)--(-0.5,0.5);
\end{tikzpicture}
```

- Lokális stílus definiálás.

Egyszerre több objektum opcióinak a beállítása

- Opciók beállításának lehetőségei.

Példa



```
\tikzset{marked/.style={orange,very thick}}  
\begin{tikzpicture}  
\draw(0,0) circle (0.5);  
\draw[marked] (-0.5,-0.5)--(0.5,0.5);  
\draw[marked] (0.5,-0.5)--(-0.5,0.5);  
\end{tikzpicture}
```

- Globális stílus definiálás.

Egyszerre több objektum opcióinak a beállítása

- Opciók beállításának lehetőségei.

Példa




```
\tikzset{marked/.style={orange,very thick}}
\begin{tikzpicture}
\draw(0,0) circle (0.5);
\draw[marked,blue] (-0.5,-0.5)--(0.5,0.5);
\draw[marked,dashed] (0.5,-0.5)--(-0.5,0.5);
\end{tikzpicture}
```

- Opciók felüldefiniálása.

Koordináták megadása

- Abszolút koordináta (x,y)

```
\tikz \draw[thick,red] (5mm,0) -- (15mm,0);
```




- Relatív koordináta +(x,y)

```
\tikz \draw[thick,red] (5mm,0) -- +(15mm,0);
```




- Névvvel rendelkező koordináta (name)

```
\tikz{\coordinate (end) at (15mm,0); \draw[thick,red] (0,0) -- (end);}
```



- Hivatkozás objektumra (obj. name)

```
\tikz{\node (A) at (1,0) {A}; \draw[thick,red] (0,0) -- (A);}
```



- Polár koordináta (szög:rádiusz)

```
\draw[thick,red] (0,0) -- (30:1);
```



Csomópontok fogalma

- TikZ node egy olyan alakzat, amihez szöveget írhatunk.

Példa



hello

```
\begin{tikzpicture}  
  \path node[shape=circle,draw,color=red] {hello};  
\end{tikzpicture}
```

- A node a path parancs része.

Csomópontok fogalma

- TikZ node egy olyan alakzat, amihez szöveget írhatunk.

Példa



hello

```
\begin{tikzpicture}  
  \node [shape=circle,draw,color=red] {hello};  
\end{tikzpicture}
```

- A node a path parancs része.
- A `\path node` parancsot röviden `\node`-ként is írhatjuk.

Csomópontok fogalma

- TikZ node egy olyan alakzat, amihez szöveget írhatunk.

Példa



hello

```
\begin{tikzpicture}  
\node[circle,draw,red] {hello};  
\end{tikzpicture}
```

- A node a path parancs része.
- A `\path node` parancsot röviden `\node`-ként is írhatjuk.
- `shape=` és `color=` opciókat el lehet hagyni.

Csomópont parancs megadása

```
\path node [opció] (név) at (koordináta) {szöveg} ...;
```

- opció vonatkozhat az alakzat formájára, színére, méretére, stb.
- A megadott névvel lehet hivatkozni a node-ra.
- A csomópont helyét az at (koordináta) segítségével adhatjuk meg. Ha hiányzik, akkor a program a path aktuális pozíciójával számol.
- A csomópontba írt szöveg tetszőleges L^AT_EX parancsokat tartalmazhat.

TikZ-ben az argumentumokat teljesen szabadon adhatjuk meg.



hello

```
\tikz \node [circle] at (0,0) [draw] (hello) [red] {hello};
```

Csomópontok elhelyezése

- A csomópontok helyét az egymáshoz viszonyított elhelyezkedésük alapján is megadhatjuk.
Például: Rajzoljuk az n_2 csomópontot az n_1 jobb oldalára.
- Relatív elhelyezés esetén a **positioning** könyvtárat kell megadni.

```
\usetikzlibrary{positioning}
```

- parancsok

- **right=of** (node ref)
- **right= 2cm of** (node ref)
- **right= 2cm and 3cm of** (node ref)

- Irányok:

- **left, right, above, below**

és ezek értelmes kombinációi pl.: **below left**

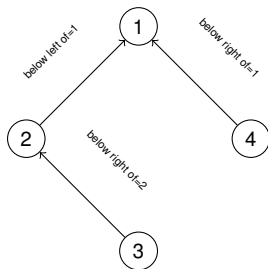
Csomópontok elhelyezése

Példa az elhelyezésre

```

\begin{tikzpicture}[draw, circle,node distance=2cm]
  \node[draw] (1) {1};
  \node[draw] (2) [below left of=1] {2};
  \node[draw] (3) [below right of=2] {3};
  \node[draw] (4) [below right of=1] {4};
\end{tikzpicture}

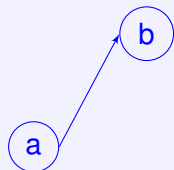
```



Csomópontok összekötése 1.

- Két csomópont kapcsolódási pontja (anchor) között tetszőleges `path` rajzolható.

Például



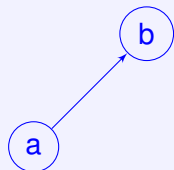
```
\begin{tikzpicture}[circle, >=latex']
\node[draw] (a) {a};
\node[draw] (b) [above right=of a] {b};
\draw[->] (a.east) -- (b.west);
\end{tikzpicture}
```

- Minden csomópont kapcsolódási pontokat tartalmaz, pl.: **east**, **west**, **center**

Csomópontok összekötése 1.

- Két csomópont kapcsolódási pontja (anchor) között tetszőleges `path` rajzolható.

Például



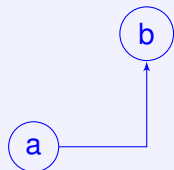
```
\begin{tikzpicture}[circle,>=latex']
\node[draw] (a) {a};
\node[draw] (b) [above right=of a] {b};
\draw[->] (a) -- (b);
\end{tikzpicture}
```

- Alapértelmezésben az alakzat középpontjába mutat a vonal iránya.

Csomópontok összekötése 1.

- Két csomópont kapcsolódási pontja (anchor) között tetszőleges `path` rajzolható.

Például



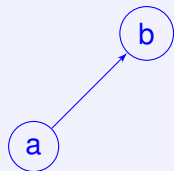
```
\begin{tikzpicture}[circle,>=latex']
\node[draw] (a) {a};
\node[draw] (b) [above right=of a] {b};
\draw[->] (a) -| (b);
\end{tikzpicture}
```

- Ortogonális összeköttetés.

Csomópontok összekötése 2.

- Két csomópontot rugalmasan a `to` `path` művelettel köthetünk össze.

Például



```

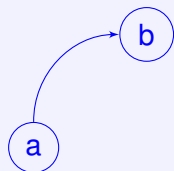
\begin{tikzpicture}[circle,>=latex']
\node[draw] (a) {a};
\node[draw] (b) [above right=of a] {b};
\draw[->] (a) to (b);
\end{tikzpicture}

```

Csomópontok összekötése 2.

- Két csomópontot rugalmasan a `to path` művelettel köthetünk össze.

Például



```

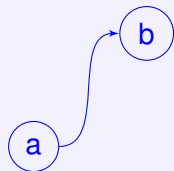
\begin{tikzpicture}[circle,>=latex']
\node[draw] (a) {a};
\node[draw] (b) [above right=of a] {b};
\draw[->] (a) to [bend left=45] (b);
\end{tikzpicture}

```

Csomópontok összekötése 2.

- Két csomópontot rugalmasan a `to` `path` művelettel köthetünk össze.

Például



```

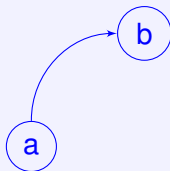
\begin{tikzpicture}[circle, >=latex']
\node[draw] (a) {a};
\node[draw] (b) [above right=of a] {b};
\draw[->] (a) to [out=0, in=180] (b);
\end{tikzpicture}

```

Összekötése feliratozása

- Összeköttetést feliratozhatjuk, ha a `path`-ba egy `node`-t szúrunk be.

Például

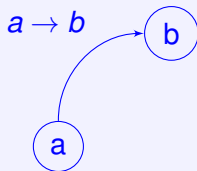


```
\begin{tikzpicture}[circle, >=latex']  
\node[draw] (a) {a};  
\node[draw] (b) [above right=of a] {b};  
\draw[->] (a) to [bend left=45] (b);  
\end{tikzpicture}
```

Összekötése feliratozása

- Összeköttetést feliratozhatjuk, ha a `path`-ba egy `node`-t szúrunk be.

Például



```

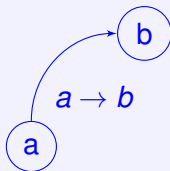
\begin{tikzpicture}[circle, >=latex']
\node[draw] (a) {a};
\node[draw] (b) [above right=of a] {b};
\draw[->] (a) to [bend left=45]
    node [auto] {a → b} (b);
\end{tikzpicture}

```

Összekötése feliratozása

- Összeköttetést feliratozhatjuk, ha a `path`-ba egy `node`-t szúrunk be.

Például



```

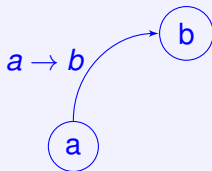
\begin{tikzpicture}[circle, >=latex']
\node[draw] (a) {a};
\node[draw] (b) [above right=of a] {b};
\draw[->] (a) to [bend left=45]
    node [auto, swap] {a → b} (b);
\end{tikzpicture}

```


Összekötése feliratozása

- Összeköttetést feliratozhatjuk, ha a `path`-ba egy `node`-t szúrunk be.

Például



```

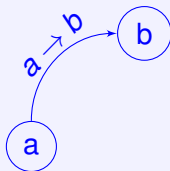
\begin{tikzpicture}[circle, >=latex']
\node[draw] (a) {a};
\node[draw] (b) [above right=of a] {b};
\draw[->] (a) to [bend left=45]
    node [left]{a → b} (b);
\end{tikzpicture}

```

Összekötése feliratozása

- Összeköttetést feliratozhatjuk, ha a `path`-ba egy `node`-t szúrunk be.

Például



```

\begin{tikzpicture}[circle, >=latex']
\node[draw] (a) {a};
\node[draw] (b) [above right=of a] {b};
\draw[->] (a) to [bend left=45]
  node [sloped, pos=0.5,
  xshift=0.1cm, yshift=0.2cm] {a → b}
  (b);
\end{tikzpicture}

```

Ismétlődő elemek rajzolása

- Létezik ciklusszervező utasítás, mellyel ismétlődő objektumokat tudunk kirajzolni.
- A ciklusokat egymásba lehet ágyazni.

Például



```
\begin{tikzpicture}
\foreach \x in {1,2,3} {
\draw [blue, ultra thick] (\x,0) circle [radius=0.3];
}
\end{tikzpicture}
```

Ismétlődő elemek rajzolása

- **foreach** ciklusváltozója lehet pár, hármas, többes érték is.
- Az értékeket / karakterrel kell elválasztani.

Gráf rajzolás

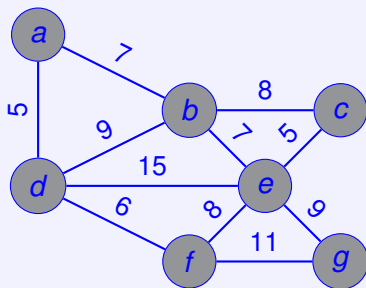
```

\tikzstyle{vertex}=[draw,circle,fill=Gray,
                    minimum size=20pt]
\tikzstyle{edge}=[draw ,thick ,-]
\tikzstyle{weight}=[font=\small,sloped, above]
\foreach \pos /\name in {(0,2)/a},{(2,1)/b},{(4,1)/c},
                      {(0,0)/d},{(3,0)/e},{(2,-1)/f},{(4,-1)/g}
\node[vertex] (\name) at \pos {$\name$};
\foreach \source /\dest /\weight in {b/a/7,c/b/8,d/a/5,
                                     d/b/9,e/b/7,e/c/5,e/d/15,f/d/6,f/e/8,g/e/9,g/f/11}
\path[edge] (\source)--node[weight] {$\weight$} (\dest);

```

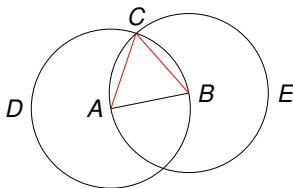
Ismétlődő elemek rajzolása

Gráf rajzolás



Metszet számítás

A metszet kiszámításához a `\usetikzlibrary{calc,through}` könyvtárakra van szükség.




```

\begin{tikzpicture}
\coordinate [label=left:$A$] (A) at (0,0);
\coordinate [label=right:$B$] (B) at (1.25,0.25);
\draw (A) -- (B);
\node (D) [draw,circle through=(B),label=left:$D$] at (A) {};
\node (E) [draw,circle through=(A),label=right:$E$] at (B) {};
\coordinate[label=above:$C$] (C) at (intersection 2 of D and E);
\draw [red] (A) -- (C);
\draw [red] (B) -- (C);
\end{tikzpicture}

```

Áthivatkozás egyik tikz környezetből a másikra

- Első lépésként globálisan meg kell adni a `\tikzstyle{every picture}+=[remember picture]` parancsot.
 - Meg kell adni a `paths` parancsra az `overlay` opciót.
- 
- A fent definiált "paths", és ez is, tulajdonképpen egy TikZ node, amit a következő képen hoztunk létre: ... adni a `\tikz[baseline,inner sep=0] \node[anchor=base](n1) {paths};` parancsra ...
 - A végén egy nyilat rajzolunk `\tikz[overlay]\draw[thick,green,->] (n2) -- (n1);`
 - a `pdflatex`-et futtassuk le kétszer egymás után.

Grafikonok, függvények rajzolása

- `\usepackage{pgfplots}` csomagot kell használni.
- Az grafikon értékeit gnuplot stílusú fájl is tartalmazhatja.

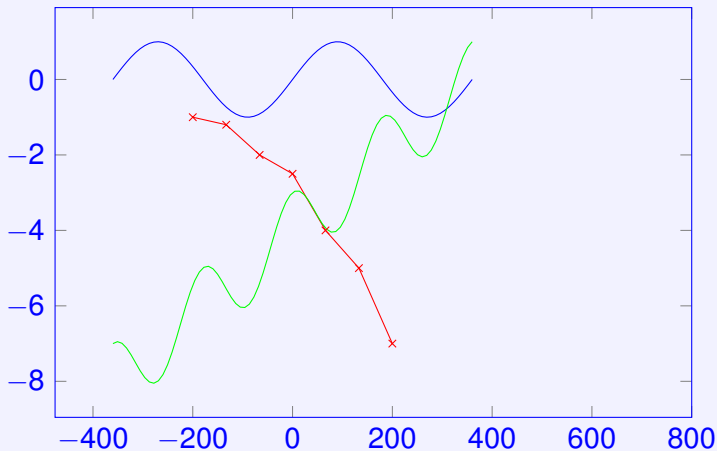
```

\begin{tikzpicture}
\begin{axis}[domain=-360:360, samples =80,
            width=10cm ,height=7cm, xmax =800]
\addplot[color=red, mark=x]
coordinates {
    (-200 , -1)  (-133 , -1.2)
    (-66 , -2)   (0 , -2.5)
    (66 , -4)    (133 , -5)
    (200 , -7)
};
\addplot[color= blue]{sin(x)};
\addplot[color= green]{-4+x /90+ cos(x *2)};
\end{axis}
\end{tikzpicture}

```


2D rajz

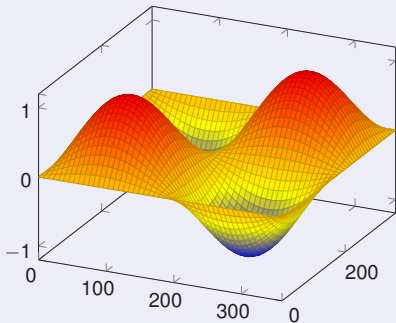
Grafikon rajzolás



Grafikonok és függvények rajzolása

3 dimenziós ábrát is készíthetünk.

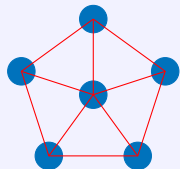
```
\begin{tikzpicture}
\begin{axis}
\addplot3[surf,domain=0:360 ,samples =50]
{sin(x)* sin(y )};
\end{axis}
\end{tikzpicture}
```



COGA embléma

Combinatorial Optimization Graph Algorithm emblémája az 5 elemű kerék.

Gráf rajzolás



```

\begin{tikzpicture}
\node [fill ,circle ,draw , RoyalBlue ] at (0 ,1) {};
\node [fill ,circle ,draw , RoyalBlue ] at (-0.9511 ,0.3091) {};
\node [fill ,circle ,draw , RoyalBlue ] at (-0.5878 , -0.8091) {};
\node [fill ,circle ,draw , RoyalBlue ] at (0.5878 , -0.8091) {};
\node [fill ,circle ,draw , RoyalBlue ] at (0.9511 ,0.3091) {};
\node [fill ,circle ,draw , RoyalBlue ] at (0 ,0) {};
\draw [red ] (0 ,1) to (-0.9511 ,0.3091) to (-0.5878 , -0.8091)
to (0.5878 , -0.8091) to (0.9511 ,0.3091) to (0 ,1);
\draw [red ] (0 ,0) to (0 ,1) (0 ,0) to (-0.9511 ,0.3091) (0 ,0)
to (-0.5878 , -0.8091) (0 ,0) to (0.5878 , -0.8091) (0 ,0)
to (0.9511 ,0.3091);
\end{tikzpicture}

```

COGA embléma kevesebb számítással

- `\pgfmathsetmacro{\x}{képlet}`
Az `\x` változó hordozza a matematikai képlet eredményét.
- `\pgfmathparse{képlet}`
az eredményt a `\pgfmathresult` fogja tartalmazni.

COGA embléma rajzolás

```

\begin{tikzpicture}
\pgfmathsetmacro {\xa }{ cos (90) }
\pgfmathsetmacro {\ya }{ sin (90) }
\pgfmathsetmacro {\xb }{ cos (90+72) }
\pgfmathsetmacro {\yb }{ sin (90+72) }
...
\node [fill ,circle ,draw , RoyalBlue ] (1) at (\xa ,\ya ) {};
\node [fill ,circle ,draw , RoyalBlue ] (2) at (\xb ,\yb ) {};
...
\draw [red ] (1) to (2) to (3) to (4) to (5) to (1);
\draw [red ] (0) to (1) (0) to (2) (0) to (3) (0) to (4) ...
\end{tikzpicture}

```

COGA embléma rajzolás polár koordináták segítségével

COGA embléma rajzolás

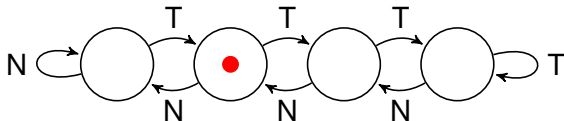
```

\begin{tikzpicture}
\node[fill,circle,draw,RoyalBlue] (1) at (90+0*72:1){};
\node[fill,circle,draw,RoyalBlue] (2) at (90+1*72:1){};
\node[fill,circle,draw,RoyalBlue] (3) at (90+2*72:1){};
\node[fill,circle,draw,RoyalBlue] (4) at (90+3*72:1){};
\node[fill,circle,draw,RoyalBlue] (5) at (90+4*72:1){};
\node[fill,circle,draw,RoyalBlue] (0) at (0 ,0) {};
\draw[red] (1) to (2) to (3) to (4) to (5) to (1);
\draw[red] (0) to (1) (0) to (2) (0) to (3) (0) to (4) ...
\end{tikzpicture}

```

Feladatok

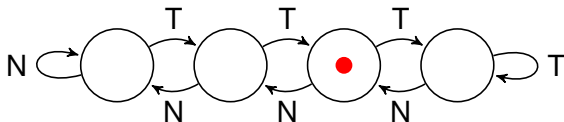
- 1 Rajzoljon egy állapotgépet, amely hasonlít a lenti ábrára.



- 2 A második, harmadik és negyedik állapotot foreach paranccsal hozza létre.
- 3 Valósítsa meg a következő dián látható folyamatábrát.

Feladatok

- 1 Rajzoljon egy állapotgépet, amely hasonlít a lenti ábrára.



- 2 A második, harmadik és negyedik állapotot foreach paranccsal hozza létre.
- 3 Valósítsa meg a következő dián látható folyamatábrát.

Feladatok

