

# *TikZ* gyorstalpaló, példák

2022. szeptember 27.

# Tartalomjegyzék

<b>1. TikZ gyorstalpaló</b>	<b>2</b>
1.1. Alapok	2
1.2. Sokszögek rajzolása, for ciklusok	5
1.3. Rácsok, szöveg beillesztése	6
1.4. Gráfok	8
<b>2. Példák</b>	<b>12</b>
2.1. Sakktábla	12
2.2. Óxisz szigete	13
2.3. Simson-egyenes	15
2.4. Komplex egységgyökök	17
2.5. Sok körzözés	19
2.6. Trigonometrikus függvények	20
2.7. Nyolcszög, lyukkal	22
2.8. KöMaL B.5131.	24
2.9. KöMaL B.5186.	25
<b>3. Impresszum</b>	<b>27</b>
3.1. Linkek	27
3.2. Készítők	27

# 1. fejezet

## TikZ gyorstalpaló

### 1.1. Alapok

A `\usepackage {tikzpicture}` kell a library implementálásához A `\begin {tikzpicture}` és `\end {tikzpicture}` parancsok közé kell helyezni a rajzolandó ábrát. A TikZ úgy működik, mint egy rajztábla. Egyesével kell az objektumokat rárajzolni, esetenként egy ciklusban többet is lehet egyszerre (lásd lejjebb). **Minden parancsot egy ; -vel kell lezárni.**

A `\begin {tikzpicture} ["paraméterek"]` ebben a szögletes zárójelben kell megadni a rajztábla paramétereit. Ilyenek:

- `scale = 3` – a képet nyújtja, kivéve a betű méretet
- `xscale = 4, yscale = 5` – ugyanez, csak merőlegesen affin képet ad

A rajzolásra két különböző, de általában mindenre elég parancs a `\draw` és `\filldraw`. A sima rajzolás csak körvonalat rajzol, a másik pedig automatikusan ugyanazzal a színnel kitölti az alakzatot. Mindkettő parancsuk meg kell mondani, hogy:

- Hova: `(x, y)`, `(fok:hossz)`
- Mit: `node`, `--` (edge), `circle`, `rectangle`, `arc`
- Stílusban: `[color, ultra thin, fill]` – ez lehet üres, ilyenkor a rajztábla stílusát használja

A node-ok kicsit trükkösebbek, róluk a gráfok részben lehet részletesebben olvasni.

```

\begin{tikzpicture}[scale=3]
  %a köröknek a kp.-át és sugarát kell megadni
  \draw (0,0) circle (0.4 cm) [color = blue!90];
  \filldraw (1,0) circle (0.4 cm) [color = red!90];

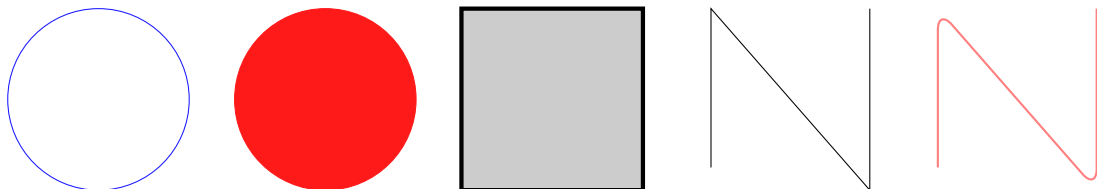
  %a téglalapoknak a balalsó és jobbfelső csúcsait kell
  megadni
  \draw (2-0.4, -.4) rectangle (2+0.4, .4) [ultra thick,
  fill=black!20];

  %a törött vonalakat csúcsról csúcsra kell megadni
  \draw (3-0.3, -0.3) -- (3-0.3, 0.4) -- (3+0.4, -0.4) --
  (3+0.4, 0.4);

  %ami sokkal menőbb, például egy rácsbejáráshoz az
  ívelt vonalak
  \draw[thick,rounded corners=8pt, color=pink!200] (4-0.3,
  -0.3) -- (4-0.3, 0.4)
  -- (4+0.4, -0.4) -- (4+0.4, 0.4);

  %Ha a törött vonalat lezárnád érdemes a --cycle befejezést
  írni a kezdő csúcs
  %megismétlése helyett.
\end{tikzpicture}

```



### 1.1.1. Illesztés

Az első fejezetben leírtakat érdemes alkalmazni. A `\clip` parancsot érdemes használni. Nem csak arra jó, hogy kivágjuk a kép egy részét, de beállítja a kép keretét, ha azzal kezdjük. Erre persze lehet használni a `\useasboundingbox` parancsot amivel megadhatunk például egy téglalappal határolt fix keretét a képnek. Amit ezen kívül rajzoltál nem fogja megjeleníteni.

```

\begin{tikzpicture}[scale=3]
  \draw (0,0) circle (0.4 cm) [color = blue!90];
  %Itt vágunk ami azt okozza, hogy az előző kör nem sérült
  \clip (-0.3, -0.3) rectangle (5, 0.3);
  \filldraw (1,0) circle (0.4 cm) [color = red!90];
  \draw (2-0.4, -.4) rectangle (2+0.4, .4) [ultra thick,
  fill=black!20];
  %Lehet relatív megadni a távolságokat, hogy ne kelljen
  mindent papíron kiszámolni
  %Ha csak sima +-t használsz, akkor a kezdő csúcstól
  viszonyít
  \draw (3-0.3, -0.3) -- ++(0, 0.7) -- ++(0.7, -0.8) -- ++(0,
  0.8);
  \draw[thick,rounded corners=8pt, color=pink!200] (4-0.3,
  -0.3) -- (4-0.3, 0.4) -- (4+0.4, -0.4) -- (4+0.4, 0.4);
\end{tikzpicture}

```



### 1.1.2. Színek, egyebek

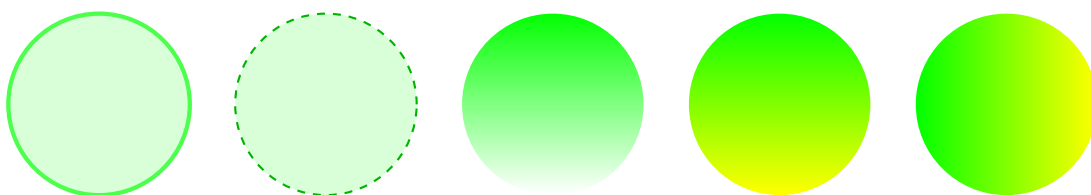
Be lehet állítani vonalvastagságot, színt és még színátmenetes ábrát is egyszerű csinálni.

- Vastagságok: { `ultra`, `very`, } + { `thin`, `thick` }
- Színek: { `red`, `green`, `blue`, `cyan`, `magenta`, `yellow`, `black`, `gray`, `darkgray`, `lightgray`, `brown`, `lime`, `olive`, `orange`, `pink`, `purple`, `teal`, `violet`, `white` }
- Vonal típusok: { `dashed`, `dotted` }
- Vonal összekötési lehetőségek (advanced):
  - `line cap = {round, rect, butt}`
  - `rounded corners = 5mm`
  - `line join = {round, bevel, mitern}`

```

\begin{tikzpicture}[scale=3]
  \draw (0,0) circle (0.4) [color = green!70, fill = green!15,
    ultra thick];
  \draw (1,0) circle (0.4) [color = green!70!black, fill =
    green!15, thick, dashed];
  \shade (2,0) circle (0.4) [top color = green];
  \shade (3,0) circle (0.4) [top color = green, bottom color =
    yellow];
  \shade (4,0) circle (0.4) [left color = green, right color =
    yellow];
\end{tikzpicture}

```



## 1.2. Sokszögek rajzolása, for ciklusok

Az, hogy lehet for ciklusokat írni, nagyban segít a valamilyen szempontból szimmetrikus ábrák elkészítésében. A for ciklusok hasonlóan más nyelvekhez bevezetnek egy változót, ami végig fut adott értékeken és végrehajtja a megadott parancsokat egyesével (jobb ha nem számít a sorrend). Lehet egymásba ágyazott ciklusokat írni, de lehet párhuzamosan két vagy több változót egyszerre változtatni. Például `\foreach \x in {1,2,3,4}{<commands>}` Ennél lehet komolyabb dolgokat is csinálni, lásd a példákat.

Eddig nem volt róla szó, de a hagyományos koordinátázás helyett lehet polárkoordinátákat is használni. `(90:1cm)` – 90 fok, 1 cm messze

A képet lehet transzformálni erre pár példa: `xshift`, `yshift`, `rotate`

```

\begin{tikzpicture}[scale = 2, ultra thick]
  \foreach \n in {3, ..., 8}{
    \draw (\n-3,0) \foreach \d
    in {1, ..., \n}{
      %MAGIC DANGER
      +(\d*360/\n:0.3cm) -- +(\d*360/\n + 360/\n:0.3cm)
    };
    %Az, hogy ilyet lehet csinálni szerintem egyszerre
    undorító és hasznos
  }
\end{tikzpicture}

```

```

%Ez kell ahhoz, hogy a szín mögé lehessen írni
változót (nem igazán lehet képletet)
\pgfmathsetmacro\i{\n*15-30}
\filldraw [xshift = \n-3, color = green!\i]
(\n-3,-1) circle (0.3cm);
}
\end{tikzpicture}

```



### 1.3. Rácsok, szöveg beillesztése

A `\draw grid` parancsot lehet négyzetrács készítésre használni a `\foreach` helyett. Meg kell adni a lépésközt és egy téglalapot ami határolja.

Szöveget beilleszteni úgy kell, hogy egy Node-ot töltsünk fel szöveggel. Paraméterként meg lehet adni, hogy az adott pozícióhoz képest, hol helyezkedjen el a csúcs és így a szöveg, ezt az `anchor=<direction>` paraméterrel lehet megadni. A `fill=white` paraméter megadásával az is elérhető, hogy a szöveg/szám alatt megszakadjanak a vonalak, így egy sokkal esztétikusabb végeredményt kapunk.

Itt különösen kiemelném a `\clip` parancs fontosságát. Ha egy ábrát szeretnénk nagyban és kicsiben is használni elég megismételni a kódot és megadunk egy keretet, ahol kíváncsiak vagyunk az ábra részleteire.

```

\begin{tikzpicture}[scale = 3]
\clip (-1.2, -0.8) rectangle (4.2,2.2); %Ez csak azért,
hogy jobban ráférjen a honlapra
%grid
\draw[step = 0.5, color=gray] (-2.1,-2.1) grid
(2.1,2.1);
%axes
\draw[->, ultra thick] (0,-2.2) -- (0,2.2);
\draw[->, ultra thick] (-2.2,0) -- (2.2,0);
%texts

```

```

\draw (0,0) [fill = white, anchor = north east] node
{\large $0$};

%y-tengely
\foreach \label in {1, 2, 3, 4}
\pgfmathsetmacro\pos{\label/2}
\draw [ultra thick](-1pt,\pos) -- (1pt, \pos) node [fill
= white, left, xshift = -7pt] {$\label$};
\foreach \label in {-1, -2, -3, -4}
\pgfmathsetmacro\pos{\label/2}
\draw [ultra thick](-1pt,\pos) -- (1pt, \pos) node [fill
= white, left, xshift = -7pt] {$\label$};

%x-tengely
\foreach \label in {1, 2, 3, 4}
\pgfmathsetmacro\pos{\label/2}
\draw [ultra thick](\pos, 1pt) -- (\pos, -1pt) node
[fill = white, below, yshift = -2pt] {$\label$};
\foreach \label in {-1, -2, -3, -4}
\pgfmathsetmacro\pos{\label/2}
\draw [ultra thick](\pos, 1pt) -- (\pos, -1pt) node
[fill = white, below, yshift = -2pt, xshift = -3pt]
{$\label$};

%ábra
\draw (1, 0.5) node [color=red, anchor = south west]
{$A$};
\draw (0.5, 1.5) node [color=blue, anchor = south west]
{$B$};
\draw (0.5,1.5) node [color=blue, circle, fill=blue,
scale =0.7] {};
\draw [->, green, dashed, ultra thick, opacity=0.5] (1,
0.5) -- (0.5, 1.5);
\draw (1, 0.5) node [color=red, circle, fill=red, scale
=0.7] {};
\draw[xshift=2.1cm, yshift=1cm] node[right,text
width=5cm]
{Az ábrán látható {\color{red} $A$} pontból megy a
{\color{blue} $B$} pontba egy {\color{green} vektor}.};
\end{tikzpicture}

```





## 1.4. Gráfok

Lehet gráfokat úgy definiálni, hogy a csúcsokat megadjuk és utána az élek már a meglévő objektumainkat (csúcsok) kössék össze. Ez azért hasznos, mert rugalmasabb lesz az ábra. Ha esetleg változtatnánk a gráfon egy új csúcs behozásával nem kell az egész ábrát koordinátáinként átírni. Elég csak a csúcsokat áthelyezni, a többit a TikZ megcsinálja nekünk. Ami még különösen hasznos, hogy tudunk a programban a csúcsoknak nevet adni és utána ezt a nevet használni referenciaként, hogy egy sokkal átláthatóbb kódot kapjunk végeredményül. Ez nem összekeverendő a csúcshoz tartozó szöveggel.

Amit szintén itt mutatnék be az a dinamikus stílus kezelés. Lehet ugyanis általunk előre definiált stílusokat megadni, hogy utána csak elég legyen annyit írni, hogy `[fontos]` vagy `[seged]`. Ezzel is azt érzük el, hogy olvashatóbb és egységesen változtathatóbb lesz a kód és így az ábránk.

A csúcsok és élek szövegezésére is sok lehetőséget ad a TikZ. A `label=<direction>:<text>` paraméter, akár többszöri használatával tudunk mindenféle szöveggel/névvel ellátni az ábránkat.

Lehet az éleket hajlítani, kigyósítani és egyéb stilisztikai trükköket alkalmazni. Erre azt ajánlom, hogy a dokumentációt érdemes olvasgatni. A következő részben írok a görbe vonalakról, ott érdemes erről olvasni.

```

\usetikzlibrary{positioning,backgrounds}
\begin{tikzpicture}[auto, node distance = 1cm and 2cm]
    \tikzstyle{StartEnd}=[rectangle,draw=blue!50,
        fill=blue!20,thick,                                inner
        sep=0pt,minimum size=6mm]

    \tikzstyle{alayer}=[circle,draw=red!80,fill=red!20,thick,
        inner sep=0pt,minimum size=6mm]

    \tikzstyle{blayer}=[circle,draw=red!80,fill=red!40,thick,
        inner sep=0pt,minimum size=6mm]
    \tikzstyle{se-edge}=[->,very thick, color=blue!30]
    \tikzstyle{in-edge}=[->,very thick, color=red!30]

    %Nodes
    \node[StartEnd] (Start) [label =
        135:\color{blue}\Large$S$] {};

    \node[alayer] (a3) [right = of Start, label=above:$a3$]
    {};
    \node[alayer] (a2) [above = of a3, label=above:$a2$] {};
    \node[alayer] (a1) [above = of a2, label=above:$a1$] {};
    \node[alayer] (a4) [below = of a3, label=above:$a4$] {};
    \node[alayer] (a5) [below = of a4, label=above:$a5$] {};

    \node[blayer] (b3) [right = of a3, label=above:$b3$] {};
    \node[blayer] (b2) [above = of b3, label=above:$b2$] {};
    \node[blayer] (b1) [above = of b2, label=above:$b1$] {};
    \node[blayer] (b4) [below = of b3, label=above:$b4$] {};
    \node[blayer] (b5) [below = of b4, label=above:$b5$] {};

    \node[StartEnd] (End) [right = of
        b3,label=45:\color{blue}\Large$C$] {};

    %Edges
    \draw[se-edge] (Start) to [out=45, in=180] (a1);
    \draw[se-edge] (Start) to [out=22.5, in=180] (a2);
    \draw[se-edge] (Start) to [out=0, in=180] (a3);
    \draw[se-edge] (Start) to [out=360-22.5, in=180] (a4);

```

```

\draw[se-edge] (Start) to [out=360-45, in=180] (a5);

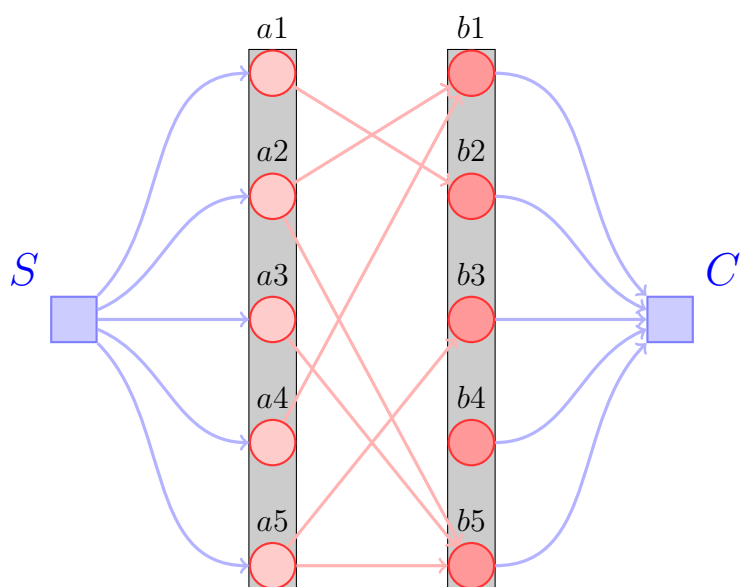
\draw[se-edge] (b1) to [out=0, in=180-45] (End);
\draw[se-edge] (b2) to [out=0, in=180-22.5] (End);
\draw[se-edge] (b3) to [out=0, in=180] (End);
\draw[se-edge] (b4) to [out=0, in=180+22.5] (End);
\draw[se-edge] (b5) to [out=0, in=180+45] (End);

\draw[in-edge] (a1) to (b2);
\draw[in-edge] (a2) to (b1);
\draw[in-edge] (a2) to (b5);
\draw[in-edge] (a3) to (b5);
\draw[in-edge] (a4) to (b1);
\draw[in-edge] (a5) to (b3);
\draw[in-edge] (a5) to (b5);

%Layers
\begin{pgfonlayer}{background}
    \filldraw [fill=black!20, draw=black] (a5.south
    -| a5.west) rectangle (a1.north -| a1.east);
    \filldraw [fill=black!20, draw=black] (b5.south
    -| b5.west) rectangle (b1.north -| b1.east);
\end{pgfonlayer}

\end{tikzpicture}

```

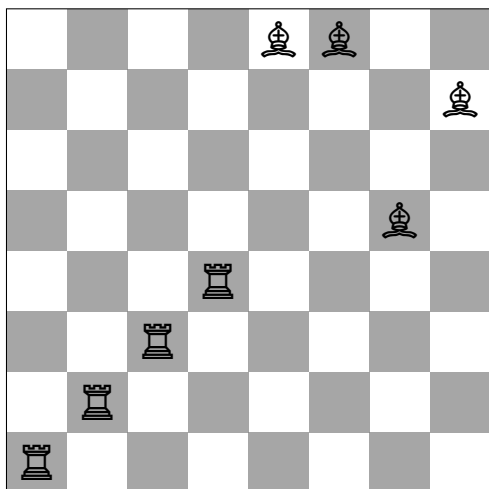


## 2. fejezet

# Példák

### 2.1. Sakktábla

```
\begin{tikzpicture}[scale=0.8]
  \foreach \i in {1,3,5,7}
  \foreach \j in {1,3,5,7}
  {
    \fill[line width=0.pt, fill=gray,opacity=0.7]
      (\i,\j) -- (\i+1,\j) -- (\i+1,\j+1) -- (\i,\j+1) --
      cycle;
    \fill[line width=0.pt, fill=gray,opacity=0.7]
      (\j,\i) --
      (\j-1,\i) -- (\j-1,\i-1) -- (\j,\i-1) -- cycle;
  }
  \draw (0,0)--(0,8)--(8,8)--(8,0)--cycle;
  \begin{Large}
    \draw (0.5,0.5) node {\bf{\symrook}};
    \draw (1.5,1.5) node {\bf{\symrook}};
    \draw (2.5,2.5) node {\bf{\symrook}};
    \draw (3.5,3.5) node {\bf{\symrook}};
    \draw (4.5,7.5) node {\bf{\symbishop}};
    \draw (5.5,7.5) node {\bf{\symbishop}};
    \draw (6.5,4.5) node {\bf{\symbishop}};
    \draw (7.5,6.5) node {\bf{\symbishop}};
  \end{Large}
\end{tikzpicture}
```

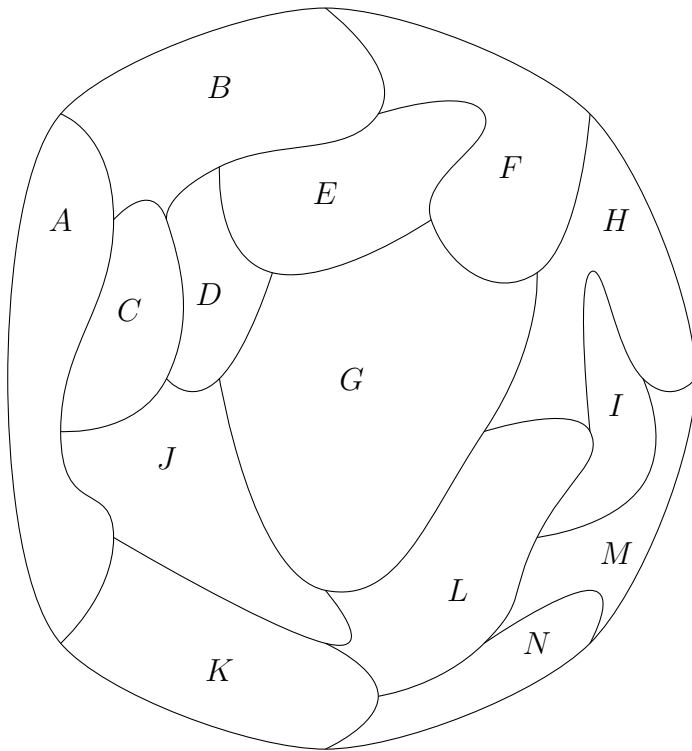


```

\draw plot [smooth, tension = 1] coordinates {(6,-1)
(8,0) (9,2) (10,4) (8,4)}; %K9
\draw plot [smooth, tension = 1] coordinates {(10,0)
(10,1) (8,0)}; %K8
\draw plot [smooth, tension = 1] coordinates {(10,4)
(10,7) (11,5) (12,5)}; %K5
\draw plot [smooth, tension = 1] coordinates {(11,5)
(11,3) (9,2)}; %K6 és K7

\draw (0,8) node {$A$};
\draw (3,10.5) node {$B$};
\draw (1.3,6.3) node {$C$};
\draw (2.8,6.6) node {$D$};
\draw (5,8.5) node {$E$};
\draw (8.5,9) node {$F$};
\draw (5.5,5) node {$G$};
\draw (10.5,8) node {$H$};
\draw (10.5,4.5) node {$I$};
\draw (2,3.5) node {$J$};
\draw (3,-0.5) node {$K$};
\draw (7.5,1) node {$L$};
\draw (10.5,1.7) node {$M$};
\draw (9,0) node {$N$};
\end{tikzpicture}

```



## 2.3. Simson-egyenes

```
\begin{tikzpicture}[line cap=round,line
join=round,>=triangle
45,x=0.7cm,y=0.7cm]
\clip(-7,-5.5) rectangle (9,5.5);
\fill[color=brown,fill=brown,fill opacity=0.1] (-3,-4)
-- (3,-4) --
(-2,4.58) -- cycle;
\draw(0,0) circle (3.5cm);
\draw [color=brown] (-3,-4)-- (3,-4);
\draw [color=brown] (3,-4)-- (-2,4.58);
\draw [color=brown] (-2,4.58)-- (-3,-4);

\draw [line width=0.4pt,domain=-7.44:9.29]
plot(\x,{(-24-0*\x)/6});
\draw [line width=0.4pt,domain=-7.44:9.29]
plot(\x,{(-5.75--8.58*\x)/-5});
```



```

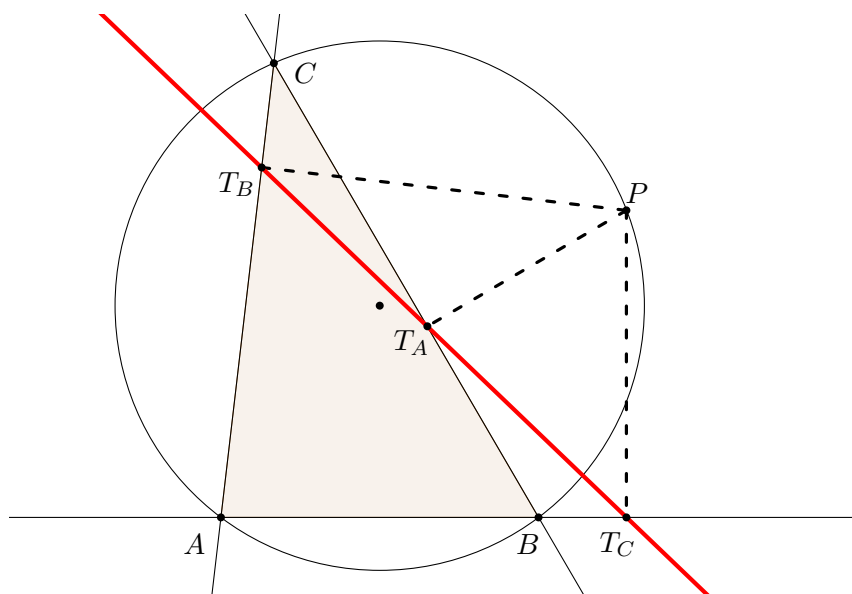
\draw [line width=0.4pt,domain=-7.44:9.29]
plot(\x,{(-21.74-8.58*\x)/-1});

\draw [line width=1.2pt,dash pattern=on 3pt off 6pt]
(4.66,1.8)-- (0.9,-0.39);
\draw [line width=1.2pt,dash pattern=on 3pt off 6pt]
(-2.23,2.61)-- (4.66,1.8);
\draw [line width=1.2pt,dash pattern=on 3pt off 6pt]
(4.66,1.8)-- (4.66,-4);

\draw [line width=1.6pt,color=red,domain=-7.44:9.29]
plot(\x,{(-1.47--3*\x)/-3.13});

\begin{small}
\fill (0,0) circle (1.5pt);
\fill (-3,-4) circle (1.5pt);
\draw (-3.5,-4.5) node {$A$};
\fill (3,-4) circle (1.5pt);
\draw (2.8,-4.5) node {$B$};
\fill (-2,4.58) circle (1.5pt);
\draw (-1.4,4.4) node {$C$};
\fill (4.66,1.8) circle (1.5pt);
\draw (4.86,2.12) node {$P$};
\fill (0.9,-0.39) circle (1.5pt);
\draw (0.6,-0.7) node {$T_A$};
\fill (-2.23,2.61) circle (1.5pt);
\draw (-2.7,2.3) node {$T_B$};
\fill (4.66,-4) circle (1.5pt);
\draw (4.5,-4.5) node {$T_C$};
\end{small}
\end{tikzpicture}

```



## 2.4. Komplex egységgyökök

### 2.4.1. Harmadik egységgyökök

```

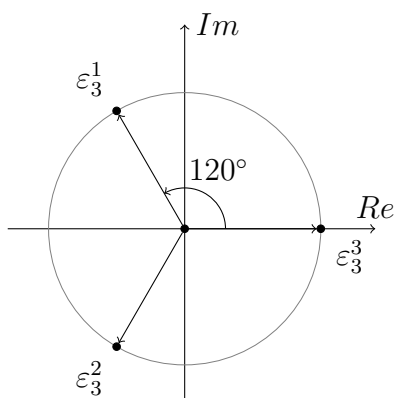
\def\n{3}
\begin{tikzpicture}[scale=1.8,
  dot/.style={draw,fill,circle,inner sep=1pt}]

\draw[>-] (-1.3,0) -- (1.4,0) node[above] {$\text{Re}$};
\draw[>-] (0,-1.3) -- (0,1.5) node[right] {$\text{Im}$};
\draw[help lines] (0,0) circle (1);

\node[dot] (0) at (0,0) {};
\foreach \i in {1,...,\n}
{
  \node[dot,label={\i*360/\n-
    (\i==\n)*45:$\varepsilon_{\n}^{\i}$}]
    (w\i)
    at (\i*360/\n:1) {};
  \draw[>-] (0) -- (w\i);
}
\draw[>-] (0:.3) arc (0:360/\n:.3);
\node at (360/\n/2:.5) {$120^\circ$};

```

```
\end{tikzpicture}
```

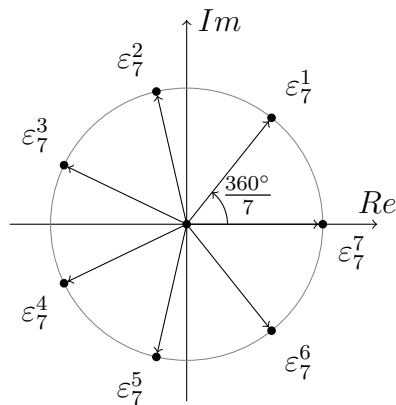


## 2.4.2. Hetedik egységgyökök

```
\def\n{7}
\begin{tikzpicture}[scale=1.8,
dot/.style={draw,fill,circle,inner sep=1pt}]

\draw[>-] (-1.3,0) -- (1.4,0) node[above] {$Re$};
\draw[>-] (0,-1.3) -- (0,1.5) node[right] {$Im$};
\draw[help lines] (0,0) circle (1);

\node[dot] (0) at (0,0) {};
\foreach \i in {1,...,\n} {
\node[dot,label={\i*360/\n-
(\i==\n)*45:$\varepsilon_{\n}^{\i}$}]
(w\i)
at (\i*360/\n:1) {};
\draw[>-] (0) -- (w\i);
}
\draw[>-] (0:.3) arc (0:360/\n:.3);
\node at (360/\n/2:.5) {$\frac{360^\circ}{\n}$};
\end{tikzpicture}
```



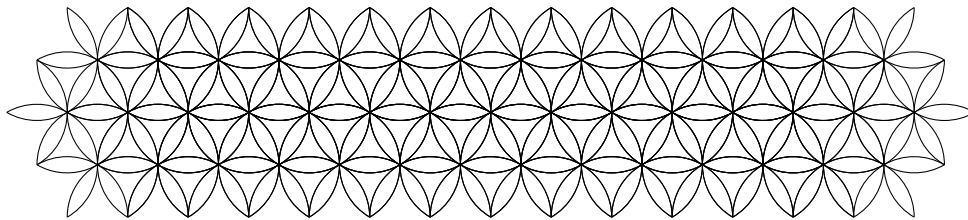
## 2.5. Sok körzőzés

```
\begin{tikzpicture}[scale=0.8]
  \foreach \i in {0,...,12}
  {
    \draw (\i+0,0) circle (1);
    \draw (\i+0.5,-0.866) arc (0:120:1);
    \draw (\i+1.5,-0.866) arc (0:240:1);
    \draw (\i+-1,-0) arc (60:-60:1);
    \draw (\i+0,-1.732) arc (0:120:1);
    \draw (\i+-1.5,-0.866) arc (-120:120:1);
    \draw (\i+-1,-0) arc (-60:60:1);
    \draw (\i+-1.5,0.866) arc (-120:0:1);
    \draw (\i+0,1.732) arc (120:360:1);
    \draw (\i+0.5,0.866) arc (0:-120:1);
    \draw (\i+0.5,0.866) arc (-60:-180:1);
    \draw (\i+0.5,0.866) arc (120:240:1);
    \draw (\i+0.5,0.866) arc (180:300:1);
    \draw (\i+0.5,-0.866) arc (180:60:1);
    \draw (\i+-1,-1.732) arc (180:60:1);
    \draw (\i+1.5,0.866) arc (120:240:1);
    \draw (\i+0.5, 0.866) arc (0:60:1);
    \draw (\i+0.5, 0.866) arc (240:180:1);
    \draw (\i+0.5, 0.866) arc (240:300:1);
    \draw (\i+0.5, 0.866) arc (120:60:1);
    \draw (\i+0.5, 0.866) arc (180:120:1);
    \draw (\i+0.5, 0.866) arc (300:360:1);
  }
\end{tikzpicture}
```

```

\draw (\i+-1,-0) arc (0:60:1);
\draw (\i+-1,-0) arc (240:180:1);
\draw (\i+-1,-0) arc (60:120:1);
\draw (\i+-1,-0) arc (300:240:1);
\draw (\i+-1,-0) arc (120:180:1);
\draw (\i+-1,-0) arc (360:300:1);
\draw (\i+0.5, -0.866) arc (180:240:1);
\draw (\i+0.5, -0.866) arc (60:0:1);
\draw (\i+0.5, -0.866) arc (120:60:1);
\draw (\i+0.5, -0.866) arc (240:300:1);
\draw (\i+0.5, -0.866) arc (120:180:1);
\draw (\i+0.5, -0.866) arc (360:300:1);
}
\end{tikzpicture}

```



## 2.6. Trigonometrikus függvények

```

\definecolor{dgreen}{rgb}{0,0.4,0}
\begin{tikzpicture}[line cap=round,line
join=round,>=triangle
45,x=1.0cm,y=1.0cm]
\draw [color=gray,dash pattern=on 2pt off 2pt,
xstep=1.5707963267948966cm,ystep=1.0cm]
(-3.89,-2.97) grid (9.33,2.94);
\draw[->,color=black] (-3.89,0) -- (9.33,0);
\draw[shift={(-3.14,0)},color=black] (0pt,2pt) --
(0pt,-2pt)
node[below] {\footnotesize $\pi$};
\draw[shift={(-1.57,0)},color=black] (0pt,2pt) --
(0pt,-2pt)
node[below] {\scriptsize $\pi/2$};

```

```

\draw[shift={(1.57,0)},color=black] (0pt,2pt) --
(0pt,-2pt) node[below]
{\footnotesize $\frac{\pi}{2}$};
\draw[shift={(pi,0)},color=black] (0pt,2pt) --
(0pt,-2pt) node[below]
{\footnotesize $\pi$};
\draw[shift={(4.71,0)},color=black] (0pt,2pt) --
(0pt,-2pt) node[below]
{\footnotesize $\frac{3}{2}\pi$};
\draw[shift={(6.28,0)},color=black] (0pt,2pt) --
(0pt,-2pt) node[below]
{\footnotesize $2\pi$};
\draw[shift={(7.85,0)},color=black] (0pt,2pt) --
(0pt,-2pt) node[below]
{\footnotesize $\frac{5}{2}\pi$};

\draw[->,color=black] (0,-2.97) -- (0,2.94);
\foreach \y in {-2,-1,1,2}
\draw[shift={(0,\y)},color=black] (2pt,0pt) --
(-2pt,0pt) node[left] {\footnotesize $\y$};
\draw[color=black] (0pt,-10pt) node[right]
{\footnotesize $0$};

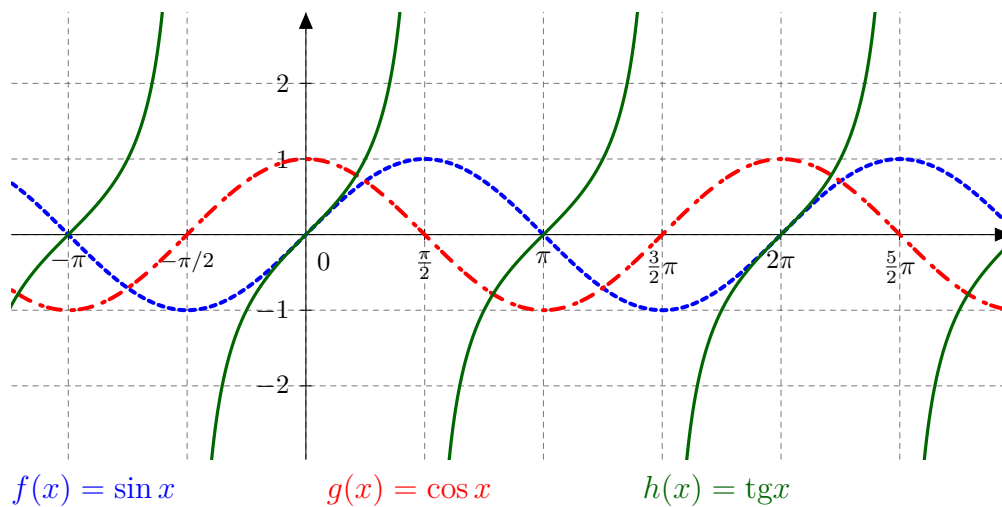
\clip(-3.89,-2.97) rectangle (9.33,2.94);
\draw[line width=1.5pt,dash pattern=on 2pt off
2pt,color=blue,
smooth,samples=100,domain=-
3.8859126567579696:9.331288233648893]
plot(\x,{sin((\x)*180/pi)});
\draw[line width=1.5pt,dash pattern=on 1pt off 2pt on
5pt off
4pt,color=red,
smooth,samples=100,domain=-
3.8859126567579696:9.331288233648893]
plot(\x,{cos((\x)*180/pi)});
\draw[line width=1.2pt, color=dgreen,
smooth,samples=100,domain=-1.56-pi:1.56-pi] plot
(\x,{sin((\x)*180/pi)/cos((\x)*180/pi)});
\draw[line width=1.2pt, color=dgreen,
smooth,samples=100,domain=-1.56:1.56] plot

```

```

(\x,{sin((\x))*180/pi)/cos((\x))*180/pi));
\draw[line width=1.2pt, color=dgreen,
      smooth,samples=100,domain=-1.56+pi:1.56+pi] plot
(\x,{sin((\x))*180/pi)/cos((\x))*180/pi));
\draw[line width=1.2pt, color=dgreen,
      smooth,samples=100,domain=-1.56+pi+pi:1.56+pi+pi]
      plot
(\x,{sin((\x))*180/pi)/cos((\x))*180/pi));
\draw[line width=1.2pt, color=dgreen,
      smooth,samples=100,domain=-
      1.56+pi+pi+pi:1.56+pi+pi+pi]
      plot
(\x,{sin((\x))*180/pi)/cos((\x))*180/pi));
\begin{scriptsize}
\end{scriptsize}
\end{tikzpicture}
\\
$\color{blue}{f(x) = \sin x} \hspace{2 cm} \color{red}{g(x) = \cos x} \hspace{2 cm} \color{dgreen}{h(x) = \mathrm{tg} x} $

```



## 2.7. Nyolcszög, lyukkal

```

\newcommand*\st{1.414142135}
\begin{tikzpicture}[scale=2, line cap=round]
  \fill[gray, pattern = horizontal lines]

```

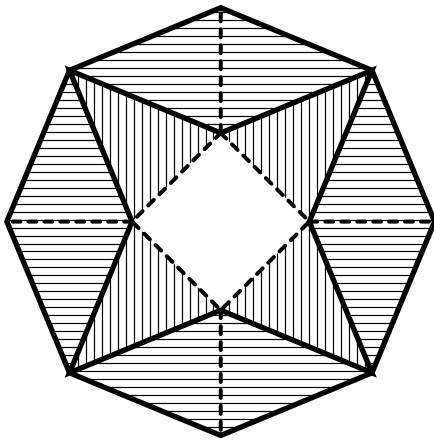
```

(-1,-1)--(0,-\st)--(1,-
1)--(\st,0)--(1,1)--(0,\st)--(-
1,1)--(-\st,0)--cycle;
\fill[white]
(-1,-1)--(0,-2+\st)--(1,-
1)--(2-\st,0)--(1,1)--(0,2-\st)--(-1,1)--(-
2+\st,0)--cycle;
\fill[gray, pattern = vertical lines]
(-1,-1)--(0,-2+\st)--(1,-
1)--(2-\st,0)--(1,1)--(0,2-\st)--(-1,1)--(-
2+\st,0)--cycle;
\fill[white]
(0,2-\st)--(2-\st,0)--(0,\st-2)--(\st-2,0)--cycle;

\draw[line width=2] (-1,-1)--(0,-\st)--(1,-
1)--(\st,0)--(1,1)--(0,\st)--(-1,1)--(-\st,0)--cycle;
\draw[line width=2] (-1,-1)--(0,-2+\st)--(1,-
1)--(2-\st,0)--(1,1)--(0,2-\st)--(-1,1)--(-
2+\st,0)--cycle;
\draw[line width=1.5, dashed]
(0,2-\st)--(2-\st,0)--(0,\st-2)--(\st-2,0)--cycle;

\draw[line width=1.5, dashed] (0,2-\st) -- (0,\st);
\draw[line width=1.5, dashed] (2-\st,0) -- (\st,0);
\draw[line width=1.5, dashed] (0,-2+\st) -- (0,-\st);
\draw[line width=1.5, dashed] (-2+\st,0) -- (-\st,0);
\end{tikzpicture}

```





## 2.8. KöMaL B.5131.

```
\begin{tikzpicture}[yscale=1.732,scale=0.7]
  \draw[dashed] (-5,0) -- (3.2,0);
  \draw[dashed] (2.2,2.2) -- (-2.5,-2.5);
  \draw[dashed] (-2.2,2.2) -- (2.5,-2.5);

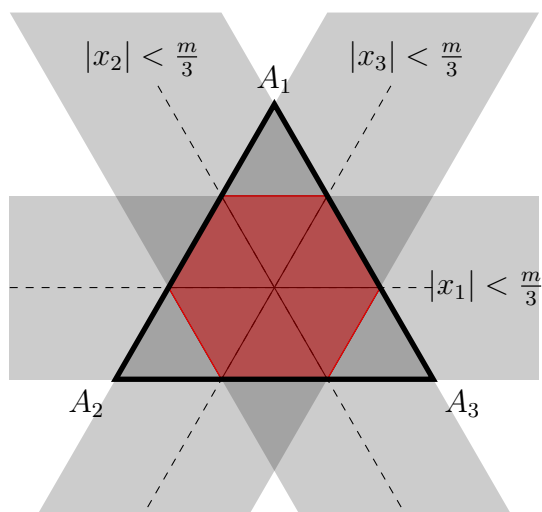
  \draw (-2,0)--(2,0); \draw (-1,1)--(1,1);
  \draw (-1,-1)--(1,1); \draw (1,-1)--(2,0);
  \draw (1,-1)--(-1,1); \draw (-1,-1)--(-2,0);

  \fill[opacity=0.2]
    (-5,-1)--(5,-1)--(5,1)--(-5,1)--cycle;
  \fill[opacity=0.2]
    (-1,3)--(-5,3)--(0.5,-2.5)--(4.5,-2.5)--cycle;
  \fill[opacity=0.2]
    (1,3)--(5,3)--(-0.5,-2.5)--(-4.5,-2.5)--cycle;

  \filldraw[red, fill opacity=0.4]
    (-1,-1)--(1,-1)--(2,0)--(1,1)--(-1,1)--(-2,0)--cycle;

  \draw[line width=2] (-3,-1)--(3,-1)--(0,2)--cycle;
  \draw (0,2) node [above] {$A_1$};
  \draw (-3,-1) node [below left] {$A_2$};
  \draw (3,-1) node [below right] {$A_3$};

  \draw (4,0) node {$|x_1| < \frac{m}{3}$};
  \draw (2.5,2.5) node {$|x_3| < \frac{m}{3}$};
  \draw (-2.5,2.5) node {$|x_2| < \frac{m}{3}$};
\end{tikzpicture}
```



## 2.9. KöMaL B.5186.

```

\begin{tikzpicture}
  \foreach \y in {2,...,10}
  \foreach \x in {1,...,10}
  {
    \draw (\x,\y) node {$\x$};
  }

  \foreach \y in {1,...,9}    \draw (-1, 11-\y) node
  {$\y$.~kör};

  \foreach \y in {2,...,10}
  {
    \draw[red,line width=2] (\y-0.2, \y-0.2) -- (\y+0.2,
    \y+0.2);
    \draw[red,line width=2] (\y-0.2, \y+0.2) -- (\y+0.2,
    \y-0.2);
  }

  \foreach \y in {2,3,4,5}
  \foreach \x in {2,...,\y}
  {
    \draw[blue,line width=1.5] (2*\x-2,2*\y) circle
    (0.3);
  }

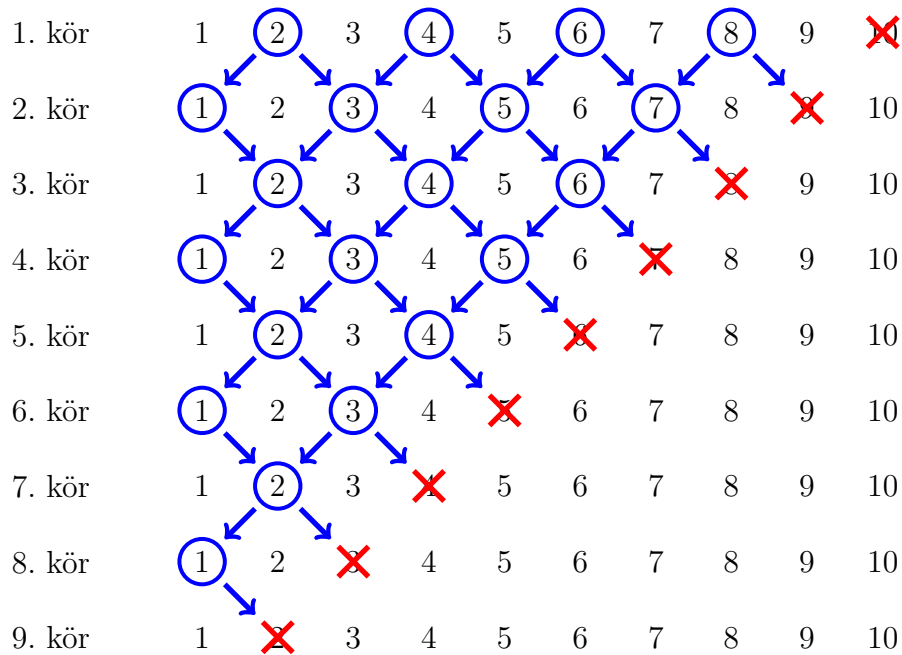
```

```

\draw[blue,line width=1.5] (2*\x-3,2*\y-1) circle
(0.3);
\draw[blue,line width=2,->] (2*\x-2.3,2*\y-0.3) --
(2*\x-2.7,2*\y-0.7);
\draw[blue,line width=2,->] (2*\x-1.7,2*\y-0.3) --
(2*\x-1.3,2*\y-0.7);
\draw[blue,line width=2,->] (2*\x-2.7,2*\y-1.3) --
(2*\x-2.3,2*\y-1.7);
}

\foreach \y in {3,4,5}
\foreach \x in {3,...,\y}
\draw[blue,line width=2,->] (2*\x-3.3,2*\y-1.3)--
(2*\x-3.7,2*\y-1.7);
\end{tikzpicture}

```



## 3. fejezet

# Impresszum

### 3.1. Linkek

- [Honlap](#)
- [PDF verzió](#)
- [Github mappa](#)
- [TikZ package \(CTAN\)](#)
- [TikZ dokumentáció \(tikz.dev\)](#)

### 3.2. Készítők

TikZ gyorstalpaló: Bertalan Dávid

Példák: Hujter Bálint

Honlap: Alexy Marcell, Szűcs Gábor

Szívesen fogadunk javaslatokat, további példákat, ezeket megírhatjátok az alapítvány [K] agondolkodasorome [P] hu címre, de küldhettek [pull requestet](#) is.