

Szövegszerkesztés – L^AT_EX

Dr. Olajos Péter

Miskolci Egyetem

Alkalmazott Matematika Tanszék

e-mail: matolaj@uni-miskolc.hu

2017/18/I.

Készült:

Wettk F., Mayer Gy., Szabó P.: L^AT_EX kézikönyv
alapján

1. Bevezetés a \LaTeX -be

- készítő:** Leslie Lamport megalkotja a \LaTeX -et
- előnyei:** magas szintű dokumentumleíró nyelv, tipográfiai tudás
- több verzió:** $\text{\LaTeX}3$, előtte $\text{\LaTeX}2_{\epsilon}$
- kezdetre egy \LaTeX állománynak:**

```
\documentclass{a dokumentumosztály neve}
preambulum [\usepackage{csomagnév}]
\begin{document}
a dokumentum szövege
\end{document}
```

1.1. Az inputenc, a magyar betűk

- más-más kódolás:** (kódkészlet: ISO-8859) operációs rendszertől és nyelvtől függően
- csomagokkal javítható:** egyik az inputenc csomag, mellyel meghatározhatjuk a szöveg kódolását
- szintaxis:** `\usepackage[latin2]{inputenc}` parancs hozzáadása a preambulumhoz, ahol a `latin2` az ISO 8859-2 szabványt jelenti, ekkor a magyar ékezetes betűk is használhatóak, pl.: „ú” lefordítódik `\’u` parancsra
- európai betűkészlet:** belső kódkészlet kiegészítése az európai nyelvek betűivel, pl.: `t1enc` (`enc=Extended Computer Modern`), melynek hatásaként az „ő” betűt vagy a `\H o`-t a fordító „ő” betűként kezeli \Rightarrow elválasztások!

- a t1enc:** további betűk is megjelentethetők, pl.: $\backslash\mathrm{DH} \Rightarrow \mathbb{D}$,
 $\backslash\mathrm{v}\{\mathrm{L}\} \Rightarrow \mathrm{L}'$, $\backslash\mathrm{k}\{\mathrm{o}\} \Rightarrow \mathfrak{o}$ stb.
- ligatúra:** ff, fi, fl stb. esetekben használjuk a $\mathrm{f}\{\mathrm{f}\}$,
 $\mathrm{f}\{\mathrm{i}\}$, $\mathrm{f}\{\mathrm{l}\}$ írásmódot
- idézőjelek:** két vessző jel alul, kettő felül \Rightarrow „magyar idézőjel”

1.2. Betűváltozatok

Példa:

abcdefghijklmnopq

`\textit{abcdefghijklmnopq}`

\Rightarrow *abcdefghijklmnopq*

`\textsl{abcdefghijklmnopq}`

\Rightarrow *abcdefghijklmnopq*

`\textsc{abcdefghijklmnopq}`

\Rightarrow ABCDEFGHIJKLMNOPQ

`\textbf{abcdefghijklmnopq}`

\Rightarrow **abcdefghijklmnopq**

`\textsf{abcdefghijklmnopq}`

\Rightarrow `abcdefghijklmnopq`

`\texttt{abcdefghijklmnopq}`

\Rightarrow `abcdefghijklmnopq`

`\textit{\texttt{kurzív írógép}}`

\Rightarrow *kurzív írógép*

–**kiemelés:** (angolul: emphasize) `\emph` parancs vagy `\em` deklarációval,

Kiemelés `\emph{normál}` szövegben \Rightarrow Kiemelés normál szövegben

`\texttt{Kiemelés {\em írógép} típusban}` \Rightarrow Kiemelés írógép stílusban

–**normál stílus:** `\textnormal` parancs és `\normalfont` deklaráció

1.3. Betűméretek

–10 betűméret:

<code>\tiny</code>	\Rightarrow	zene,
<code>\scriptsize</code>	\Rightarrow	zene,
<code>\footnotesize</code>	\Rightarrow	zene,
<code>\small</code>	\Rightarrow	zene,
<code>\normalsize</code>	\Rightarrow	zene,
<code>\large</code>	\Rightarrow	zene,
<code>\Large</code>	\Rightarrow	zene,
<code>\LARGE</code>	\Rightarrow	zene,
<code>\huge</code>	\Rightarrow	zene,
<code>\Huge</code>	\Rightarrow	zene.

–**új betűparancs előállítása:** (egy paraméteres)

```
\newcommand{parancs neve}{parancs def.}
```

(két paraméteres)

```
\newcommand{\sztr}[2]{\textnormal{#1}-\textit{#2}}
```

–**\renewcommand parancs:** átdefiniálása egy parancsnak (lásd később: Makrók)

1.4. Matematikai formulák

- matematikai mód:** használható a `$...$` ill. `$$...$$` jelek, vagy a (szövegközi) `\(...\)`, `\begin{math}...\end{math}` ill. (kiemelt) `\[...\]`, `\begin{displaymath}...\end{displaymath}`
- egyenlet(ek):** `equation`, `eqnarray` és `eqnarray*` környezetek használata

Példa:

```
\begin{equation}
1+7+12=20
\end{equation}
```

$$1 + 7 + 12 = 20 \tag{1}$$

Példa: (folytatás)

```
\begin{eqnarray}
```

```
1+8&=&9\\
```

```
1+8+7&=&16\nonumber\\
```

```
1+8+7+10&=&26
```

```
\end{eqnarray}
```

$$1 + 8 = 9 \quad (2)$$

$$1 + 8 + 7 = 16$$

$$1 + 8 + 7 + 10 = 26 \quad (3)$$

```
\begin{eqnarray*}
```

$$1+8=&9\\$$

$$1+8+7=&16\\$$

```
\noalign{és a harmadik egyenlet...}\\
```

$$1+8+7+10=&26\\$$

```
\end{eqnarray*}
```

$$1 + 8 = 9$$

$$1 + 8 + 7 = 16$$

és a harmadik egyenlet...

$$1 + 8 + 7 + 10 = 26$$

Vegyük észre a nagyobb helyközt! Ezen hiba javítása:

```
\setlength{\arraycolsep}{.13889em}
```

Más megoldás: `multline` és `split` környezetek

```
\begin{multline}
  1+8+27+64 \quad \backslash\backslash \\
  =1+3+5+7+9+11 \quad \backslash\backslash \\
  +13+15+17+19 \\
\end{multline}
```

$$1 + 8 + 27 + 64$$

$$= 1 + 3 + 5 + 7 + 9 + 11$$

$$+ 13 + 15 + 17 + 19 \quad (4)$$

```

\begin{equation}
\begin{split}
100 &= 1+8+27+64 = \{\}\backslash\backslash \\
&= 1+3+5+7+9+\{\}\backslash\backslash \\
&\quad +11+13+15+17+19
\end{split}
\end{equation}

```

$$\begin{aligned}
 100 &= 1 + 8 + 27 + 64 = \\
 &= 1 + 3 + 5 + 7 + 9 + \\
 &\quad + 11 + 13 + 15 + 17 + 19
 \end{aligned} \tag{5}$$

–**bal oldali számozás:** a

`\documentclass[leqno]{article}`-ban a `leqno` opció

–**matematikai képletekben magyarázó szöveg:** használjuk a

`\text` előtag helyett a `\math` tagot, pl.: `\mathit`,
`\mathnormal`, `\mathbf` stb.

–**kalligrafikus, gót és duplázott betűk:** `\mathcal`, `\mathfrak`
 (eufrak csomag) és `\mathbb` (amssymb) parancsok, pl.:

`\mathcal{ABCDEFGHJKLMNOP}` \Rightarrow

ABCDEF GHIJK LMNOP

`\mathfrak{ABCDEFGHJKLMNOP}` \Rightarrow

ⒶⒷⒸⒹⒺⒻⒼⒽⒿⓀⓁⓂⓃⓄⓅ

`\mathbb{ABCDEFGHJKLMNOP}` \Rightarrow

ⒶⒷⒸⒹⒺⒻⒼⒽⒿⓀⓁⓂⓃⓄⓅ

–**gyökvonás:** minden a `\sqrt` paranccsal működik, pl.:

$$\text{\texttt{\$}\sqrt{a+b}\text{\texttt{\$}}} \Rightarrow \sqrt{a+b}$$

$$\text{\texttt{\$}\sqrt[3]{2+3}\text{\texttt{\$}}} \Rightarrow \sqrt[3]{2+3}$$

Indexelés, függvénynevek hasonlóan, mint az $\mathcal{A}\mathcal{M}\mathcal{S}$ - $\text{\texttt{TeX}}$ -ben

–**kongruencia:** `\pmod`, `\bmod`, `\mod`, `\pod` alkalmazása

Példa:

$$\text{\texttt{\$a\bmod m\$}} \Rightarrow a \bmod m$$

$$\text{\texttt{\$a\equiv b\pmod m\$}} \Rightarrow a \equiv b \pmod{m}$$

$$\text{\texttt{\$a\equiv b\mod\{m\}\$}} \Rightarrow a \equiv b \pmod{m}$$

$$\text{\texttt{\$a\equiv b\pod\{m\}\$}} \Rightarrow a \equiv b \pmod{m}$$

–**mátrixok:** array környezet, pl.:

```
\[
f(x)=\left\{
\begin{array}{ll}
0 & \& \mbox{, ha } \$x\$ \text{ racionális,} \\
1 & \& \mbox{, ha } \$x\$ \text{ irracionális}
\end{array}
\right.
```

$$f(x) = \begin{cases} 0 & , \text{ ha } x \text{ racionális,} \\ 1 & , \text{ ha } x \text{ irracionális} \end{cases}$$

1.5. Szavak, mondatok, bekezdések

- bekezdések:** `\par` parancs vagy legalább egy üres sor (utóbbit használjuk!) \Rightarrow új bekezdés, `\\` pedig sortörést eredményez
- `\usepackage[magyar]{babel}`: magyar szavak elválasztási szabályai
- `\selectlanguage{nyelv}`: (pl. `english`) parancs a nyelvek közötti váltást valósítja meg
- kiemelés sorzárással:**
`\begin{flushleft}...\end{flushleft}`, hasonlóan a `flushright`, `center` opcióknál
- idézetek, versek:** `quote`, `quotation` és `verse` környezetek

Példa:

Idézetek az idézetekről:

```
\begin{quote}
```

```
  ,,Utálom az idézeteket. Azt mondd, amit  
  tudsz!’’ \emph{Ralph Waldo Emerson}
```

```
  ,,Gyakran idézek magamtól, ez  
  megfűszerezi beszélgetéseimet.’’
```

```
  \emph{George Bernard Shaw}
```

```
\end{quote}    További\dots
```

Példa: (folytatás)

Idézetek az idézetekről:

„Utálok az idézeteket. Azt mondd, amit tudsz!” Ralph
Waldo Emerson

„Gyakran idézek magamtól, ez megfűszerezi
beszélgetéseimet.” George Bernard Shaw

További...

(A quotation környezet esetén minden bekezdés esetén van
behúzás és nincs megnövelt sorköz!)

Példa: (folytatás)

```
\begin{verse}
  Megy a juhász számaron,\\
  Földig ér a lába;\\
  Nagy a legény, de nagyobb\\
  Boldogtalansága.

  Gyepes hanton furulyált,\\
  Legelészett nyája.\\
  \dots
\end{verse}
```

Példa: (folytatás)

Megy a juhász számáron,
Földig ér a lába;
Nagy a legény, de nagyobb
Boldogtalansága.

Gyepes hanton furulyált,
Legelészett nyája.

...

(A verse környezet esetén verssorok \-vel való elválasztása!)

–listák készítése: `itemize`, `enumerate`, `description`, `list`
környezetek

Példa:

A~\LaTeX\ a listatípusok mindegyikére
felkínál egy környezetet.

```
\begin{itemize}
```

```
\item Felsorolásokhoz az itemize,
```

```
\item számozott listákhoz az enumerate,
```

```
\item leíró listákhoz a description,
```

```
\item egyéni listákhoz a list  
környezetet alkalmazzuk.
```

```
\end{itemize}
```

Példa: (folytatás)

A \LaTeX a listatípusok mindegyikére felkínál egy környezetet.

- Felsorolásokhoz az `itemize`,
- számozott listákhoz az `enumerate`,
- leíró listákhoz a `description`,
- egyéni listákhoz a `list` környezetet alkalmazzuk.

Példa: (folytatás)

A `\texttt{paralist}` csomag bekezdésen belüli listát adó környezetei:

```
\begin{compactitem}[--]  
\item \texttt{compactitem} a felsorolásokhoz,  
\item \texttt{compactenum} a számozott  
listákhoz,  
\item \texttt{compactdesc} a leíró listákhoz.  
\end{compactitem}
```

A `paralist` csomag bekezdésen belüli listát adó környezetei:

- `compactitem` a felsorolásokhoz,
- `compactenum` a számozott listákhoz,
- `compactdesc` a leíró listákhoz.

Példa: (folytatás)

```
\begin{enumerate}
\item Felsorolásokhoz használható az
\texttt{itemize} környezet.
\item Sorszámozott listákhoz való az
\texttt{enumerate} környezet. A sorszámai lehetnek
  \begin{enumerate}
    \item arab számok,
    \item \begin{enumerate}
      \item kis római számok,
      \item nagy római számok,
    \end{enumerate}
  \end{enumerate}
\item kis- és nagybetűk.
\end{enumerate} \end{enumerate}
```

Példa: (folytatás)

1. Felsorolásokhoz használható az `itemize` környezet.
2. Sorszámozott listákhoz való az `enumerate` környezet. A sorszámok lehetnek
 - a) arab számok,
 - b) i. kis római számok,
ii. nagy római számok,
 - c) kis- és nagybetűk.

Példa: (folytatás)

Sorszámozott lista sorszáma

```
\begin{compactenum}[I.]
```

```
\item arab szám,
```

```
\item római szám vagy
```

```
\item betű
```

```
\end{compactenum}
```

egyenként lehet.

Sorszámozott lista sorszáma

I. arab szám,

II. római szám vagy

III. betű

egyenként lehet.

Példa: (folytatás)

inparaenum környezet a paralist csomagból:

Sorszámozott lista sorszáma

```
\begin{inparaenum}[\itshape a)]
```

```
\item arab szám,
```

```
\item római szám vagy
```

```
\item betű
```

```
\end{inparaenum}
```

egyaránt lehet.

Sorszámozott lista sorszáma *a)* arab szám, *b)* római szám vagy *c)* betű egyaránt lehet.

Példa: (folytatás)

```
\begin{description}
\item[Felsorolás] készítésére szolgál az
    \texttt{itemize} környezet.
\item[Sorszámozott lista] készíthető
    az \texttt{enumerate} környezettel.
\item[Leíró lista] képzésére való a
    \texttt{description} környezet.
\end{description}
```

Felsorolás készítésére szolgál az `itemize` környezet.

Sorszámozott lista készíthető az `enumerate` környezettel.

Leíró lista képzésére való a `description` környezet.

A list környezet: `\begin{list}{alapért}{beállítások}`,
alapért: az a címke, amely minden listaelem elején megjelenik
beállítások: minden listaelemre megadható beállítások

Példa:

Ez itt a listát megelőző szöveg, innen látszik, hogy
milyen széles a sor.

```
\begin{list} {$\rightarrow$}
  {\setlength{\leftmargin}{2em}
    \setlength{\rightmargin}{\leftmargin}}
\item Ez az első listaelem.
Ezután következik a második.
\item Ez a második listaelem. \end{list}
```

Példa: (folytatás) Ez itt a listát megelőző szöveg, innen látszik, hogy milyen széles a sor.

⇒ Ez az első listaelem. Ezután következik a második.

⇒ Ez a második listaelem.

–**tételszerű környezetek:** `\newtheorem` parancs, pl.:

`\newtheorem{thm}{Theorem}` (preambulum), utána a
`\begin{thm}...\end{thm}` használható

–`\newtheorem{thm}{Theorem}[section]`: az aktuális
 fejezetcím jelenik meg tételszámként!

Ha azt akarjuk, hogy a számozás két vagy több környezet
 esetén folyamatosan menjen, akkor

`\newtheorem{dfn}{thm}{Definition}`

1.6. Téma, oldalformátum

- papírméret:** a `\documentclass[a4paper]{article}` módon adhatjuk meg (`a5paper`, `b5paper`, `letterpaper` stb.)
- betűméret:** `\documentclass[12pt,a4paper]{article}` megadás (10, 11, 12pt-os)
- oldalformátum:** `twocolumns` és `twoside` paraméterek megadása, a szöveg közben ezek megváltoztatása:
`\twocolumns` és `\onecolumn` parancsok
- oldal stílusa:** (fejléc, lábléc, szöveg) `\pagestyle{opció}` parancs, ahol az opcionális paraméter lehet `empty`, `plain`, `headings` stb.
- oldalszámozás:** `\pagenumbering{paraméter}` parancs, ahol a paraméter lehet `arabic`, `roman`, `Roman`, `alpha`, `Alpha` stb.

1.7. Egy dokumentum részei

–**cím, szerző, dátum:**

```
\title{a cím}
```

```
\author{szerző neve}
```

```
\date{}
```

```
\maketitle,
```

ahol a `\date` után megadhatunk dátumot, vagy elhagyva az utolsó fordítás dátuma jelenik meg, külön címlap:

`titlepage` vagy `notitlepage` megadása a

`\documentclass`-ban, ill. készíthetünk saját címlapot a

`\begin{titlepage}...\end{titlepage}` segítségével,

dátum itt: `\today` parancs

–**kivonat készítése:** `\begin{abstract}...\end{abstract}`

- fejezetekre osztás:** `\chapter`, `\section`, `\subsection`,
`\subsubsection`, `\paragraph`, `\subparagraph`, automatikus
számozás elkerülése, pl.: `\section*`
- lábjegyzet, széljegyzet:** `\footnote`, `\marginpar{szöveg}`
- irodalomjegyzék:** `\begin{thebibliography}{szám}`
`\bibitem{hivatkozási név}szerző neve, kiadó neve,`
évszám stb. ... `\end{thebibliography}`, ahol a hiv. név
alapján lehet a `\cite{hiv. név}` parancs használatával
utalni egy irodalomra, dinamikus számozás

1.8. Táblázatok

–tabular környezet:

```
\begin{tabular}{lr...clr}  
elem & elem & ... & elem \\  
... \\  
\end{tabular}
```

Példa:

```
\begin{tabular}{lrrrr}  
Budapest & 7:00 & 9:30 & 13:15 \\  
Dömsöd & 7:58 & 10:40 & 14:38 \\  
\end{tabular}
```

Példa: (folytatás)

Budapest 7:00 9:30 13:15

Dömsöd 7:58 10:40 14:38

–vonal húzása (vízszintes és függőleges):

Példa:

```
\begin{tabular}{||l|rrr||}
```

```
\hline\hline Budapest & 7:00 & 9:30 & 13:15 \\\hline
```

```
Dömsöd      & 7:58 & 10:40 & 14:38 \\\
```

```
\hline\hline
```

```
\end{tabular}
```

Budapest	7:00	9:30	13:15
Dömsöd	7:58	10:40	14:38

- **A * és @ parancsok:** ismétlődő karakterlánc esetén (pl.: *5r vagy *{5}{r}) vagy tetszőleges karaktereket helyeznek el az oszlopok között (pl.: @: vagy @{:})

Példa:

```
\begin{tabular}
  {@{1 }l@{ = }r@{,}l@{\,mm}}
  pont & 0 & 35 \\
  pica & 4 & 22 \\
  inch & 25 & 4
\end{tabular}
```

1 pont = 0,35 mm

1 pica = 4,22 mm

1 inch = 25,4 mm

Példa:

```
\begin{tabular}{|cc|}
  \hline X & X \\\hline
\end{tabular}

\begin{tabular}{|@{}c@{}c@{}c@{}|}
  \hline X & X \\\hline
\end{tabular}
```

X	X	XX
---	---	----

–adott szélességű táblázat: $\{p\{2\text{ cm}\}\}$ -rel adott szélességű
oszlopok

Példa:

```
\begin{tabular*}{40mm}
{@{}l@{ -- }l@{\extracolsep{\fill}}r@{}}
\toprule
FTC    & MTK & 1:1 \\
        & Vasas & ETO & 0:0 \\
\bottomrule
\end{tabular*}
```

FTC – MTK	1:1
Vasas – ETO	0:0

–összevont oszlopok:

Példa:

```
\begin{tabular}{|l| c c|}
\cline{2-3} \multicolumn{1}{c|}{}
&\multicolumn{2}{c|}{Év}\\ \cline{2-3}
\multicolumn{1}{c|}{}&2002&2003\\ \hline
Jövedelem & 775000 & 866500\\
Adó & 165000 & 194950\\ \hline
\end{tabular}
```

	Év	
	2002	2003
Jövedelem	775000	866500
Adó	165000	194950

–összevont sorok: multirow csomag

Példa:

```
\begin{tabular}{|c|c|c|c|c|}
\hline
\multirow{2}[3]*{\textit{Input pontok}} &
\multicolumn{2}{c|}{\textit{eredeti}} &
\multicolumn{2}{c|}{\textit{dinamikus}}\\
& \textit{iteráció} & \textit{neuronok} &
\textit{iteráció} & \textit{neuronok}\\
\hline
10 & 3200 & 40 & 2300 & 30
\hline
\end{tabular}
```

<i>Input pontok</i>	<i>eredeti</i>		<i>dinamikus</i>	
	<i>iteráció</i>	<i>neuronok</i>	<i>iteráció</i>	<i>neuronok</i>
10	3200	40	2300	30

–**tabbing környezet:** szöveg tabulálása, azaz tabulátorhelyek kijelölése, \= a tabuláció helye, \> ugrás

Példa:

```
\begin{tabbing}
akkor,\= \\
      \>elvtársak, itt a tollam,\\
írjanak\= \\
      \>maguk\= \\
      \>      \>helyettem!
\end{tabbing}
```

Példa: (folytatás)

akkor,
 elvtársak, itt a tollam,
írjanak
 maguk
 helyettem!

```
\begin{tabbing}
IF \=THEN \= \kill
IF \>korán kelsz\\
    \>THEN \>aranyat lelsz\\
    \>ELSE \>nagyot alszol
\end{tabbing}
```

Példa: (folytatás)

IF korán kelsz

 THEN aranyat lelsz

 ELSE nagyot alszol

\kill parancs: a sor nem jelenik meg a nyomtatási képben, de a benne levő parancsok végrehajtódnak

1.9. Rajzok, képek, ábrák

–**kétféle lehetőség:** `picture` környezet vagy eszköz meghajtó segítségével előállítani az ábrát

`picture` **környezet:** koordinátarendszer, egység megadása:

`\unitlength{mérték}` vagy

`\setlength{\unitlength}{mérték}`, rajz mérete,

referenciapont megadása:

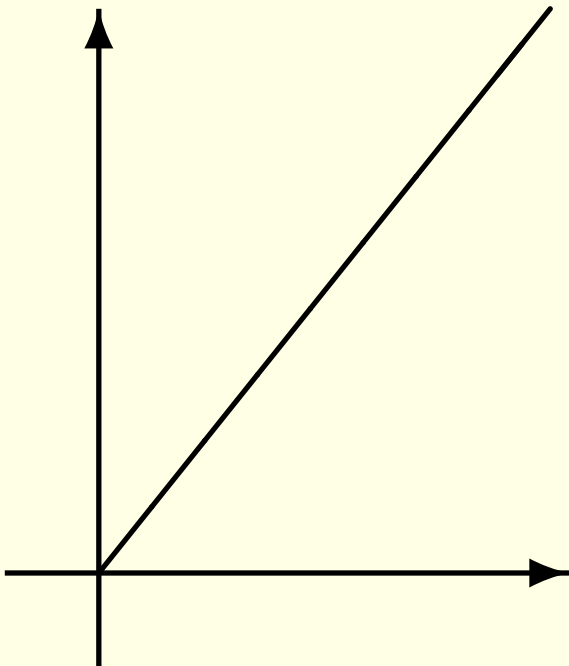
`\begin{picture}(width,height)(x origó, y origó)`

vonalvastagság: `\thicklines` (vastag), `\thinlines` (vékony)

–Egyenesek, vektorok, körök:

Példa:

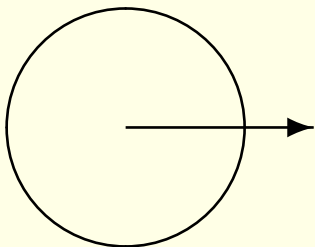
```
\unitlength 1mm  
\begin{picture}(30,35)(-5,-5)  
  \thicklines  
  \put(-5,0){\vector(1,0){30}}  
  \put(0,-5){\vector(0,1){35}}  
  \put(0,0){\line(4,5){24}}  
\end{picture}
```



```

\unitlength 1mm
\begin{picture}(50,12)
\put(6,6){\vector(1,0){10}} \put(6,6){\circle{12}}
\put(40,6){\circle*{5}}
\end{picture}

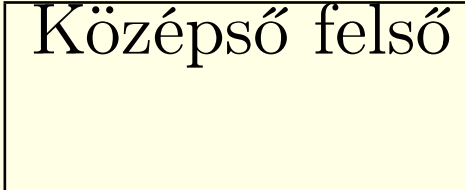
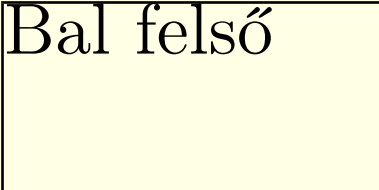
```



–**Dobozok:** `\makebox` és `\framebox` parancsok

Példa:

```
\begin{picture}(30,35)(-5,-5)
\put(0,10){\framebox(40,20)[lt]{Bal felső}}
\put(50,10){\framebox(50,20)[t]{Középső felső}}
\end{picture}
```

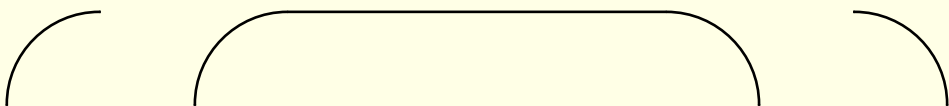


többsoros szöveg: `\shortstack[poz]{szöveg}`, sortörés `\\`-vel

–Ovális: `\oval(x,y)[részlet]`

Példa:

```
\begin{picture}(50,30)
\put(5,25){\oval(10,10)[t1]}
\put(25,25){\oval(30,10)[t]}
\put(45,25){\oval(10,10)[tr]}
\end{picture}
```

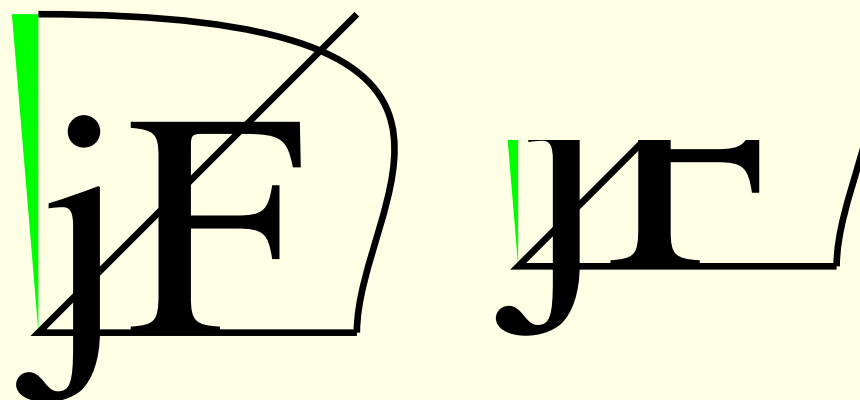


–A graphics és a graphicx csomagok:

Csak .eps formátum támogatása! A kép beszúrása:

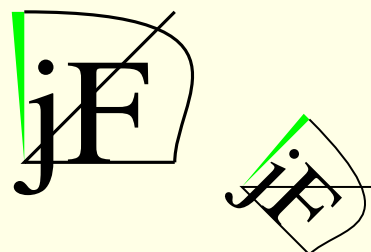
```
\includegraphics[bb=10 10 12 14]{jf}\hspace*{2cm}
```

```
\includegraphics[viewport=0 0 60 30,clip]{jf}
```



```
\includegraphics[keepaspectratio,width=3cm,  
totalheight=1cm]{jf}
```

```
\includegraphics[angle=-45,scale=0.3]{jf}
```



Az epsfig csomag

```
\epsfig{file=mikulas.eps,width=4cm, height=4.cm}
```



1.10. Jegyzékek

–**tartalomjegyzék:** `\tableofcontents`, `\index` (`\makeindex` használata)

`\setcounter{tocdepth}{szám}`

–**hivatkozások:** `\label`, `\pageref`, `\ref`, `\eqref`, `\cite`

2. Testreszabás, programozás

2.1. Osztály, stílus

- style vagy class:** 1994 előtt `\documentstyle`, utána `\documentclass` és `\usepackage` parancsok
- osztályok és csomagok:** `article`: cikkekhez; `book`: könyvek; `report`: beszámoló, értekezés; `letter`: levél írásához; `amsart`: \mathcal{AMS} -t megvalósító cikk; fóliakészítés: `slides`, `powersem` stb.

2.2. A L^AT_EX programozása

–parancs szintaxisa:

```
\parancs*[opc]{köt}...[opc]{köt}
```

–új parancs létrehozása:

```
\newcommand*{prncs}[argsz][alapért]{def}
```

```
\renewcommand*{prncs}[argsz][alapért]{def}
```

```
\providenewcommand*{prncs}[argsz][alapért]{def}
```

–Létező parancs ellenőrzése:

```
\CheckCommand*{prncs}[argsz][alapért]{def}
```

a válasz a log állományban, "Warning..." formában

–Argumentum nélküli parancsok:

Példa:

```
\newcommand*{\E}{\ensuremath{\mathbb{E}}}
```

Jelölje $\backslash E$ az euklideszi teret, legyen továbbá $a, b \in E$.

Jelölje \mathbb{E} az euklideszi teret, legyen továbbá $a, b \in \mathbb{E}$.

–Argumentumos parancsok:

Példa:

```
\newcommand*{\QD}[2]{\ensuremath{\sum_{i,j}
#1_{ij}#2_i#2_j}}
\QD{a}{x} és \QD{b}{y} értéke \dots
```

Példa: (folytatás)

$\sum_{i,j} a_{ij}x_ix_j$ és $\sum_{i,j} b_{ij}y_iy_j$ értéke ...

2.3. A L^AT_EX belső parancsai

–Az @ (at) karakter kiemelt szerepe: minden belső parancsban (primitívben) benne van, ezért ha .tex forrásban használjuk őket, akkor a

```
\makeatletter
```

```
...
```

```
\makeatother
```

közé tesszük őket.

–Előre megadott belső parancsok:

```
\z@ = 0 vagy 0pt
```

```
\p@ = 1pt
```

```
\@firstoftwo
```

```
\@firstofone
```

Példa:

```
\makeatletter \newcounter{cntr}
\@whilenum \thecntr<10 \do
  {\stepcounter{cntr}\thecntr\ }
\newcommand{\nap}{felkel, lenyugszik}
\@for \mitcsinal:=\nap \do
  {A nap \mitcsinal. }
\makeatother
```

1 2 3 4 5 6 7 8 9 10 A nap felkel. A nap lenyugszik.

–**Egymásba ágyazott definíciók:** Külső parancs argumentumaira a #1, #2,..., a belső parancs argumentumaira ##1, ##2,... módon lehet hivatkozni.

Példa:

```
\newcommand{\irogep}[1]{%
  \renewcommand{\emph}[1]{\underline{##1}}
  \texttt{#1}}
{\irogep{Ki van \emph{emelve},}
 \emph{emelve},} \emph{emelve}.
```

Ki van emelve, emelve, emelve.

–új környezet definiálása:

```
\newenvironment*{körny}[argsz][alapért]{nyitódef}
{végdef}
```

```
\renewenvironment*{körny}[argsz][alapért]{nyitódef}
{végdef}
```

–**argumentum nélküli környezetek:** amikor egy másik környezet hívódik meg, pl.:

```
\newenvironment{jobbra}
{\begin{flushright}\itshape}
{${\clubsuit}$\end{flushright}}
```

```
\begin{jobbra}
  Szerencséje lesz.
\end{jobbra}
```

Szerencséje lesz. ♣

2.4. Méretek

–**betűk mozgatása:** `\kern` parancs: a betűk közötti távolság megváltoztatása, `\lower`: lejjeb mozgat

–**hosszúságparancsok:** `\fill` ill. `\stretch{szám}`

`\newlength{\hosszparancs}` (0 pt),

`\setlength{\hosszparancs}{hossz}` (beállít),

`\addtolength{hosszparancs}{hossz}` (növel-csökken),

`\the\hosszparancs` (kiíratás pt-ben)

–**vízszintes helykihagyás:**

`\hspace*{hossz}` szóköz, `*` esetén mindig kihagyja az adott hosszt!

–függőleges helykihagyás:

`\vspace*{hossz}` függőleges helykihagyás, `*` esetén mindig kihagyja az adott hosszt!

`\addvspace{hossz}` függ. helyk., ha több ilyen van, akkor csak a legnagyobb hajtódik végre

2.5. Számlálók

–**beépített számlálók:** egész értéket tárolnak, magunk is definiálhatunk ilyeneket

pl.: **part:** rész sorszáma, **figure:** ábra sorszáma, **footnote:** lábjegyzet sorszáma stb.

–**új számláló:**

`\newcounter{számláló}{számlálóős}`: kezdőérték 0, ha a számlálóős megváltozik, akkor ismét 0 lesz, megváltoztatásuk:

`\setcounter{számláló}{szám}`

`\addtocounter{számláló}{szám}`

`\stepcounter{számláló}`

`\refstepcounter{számláló}`

–kiíratás:

`\arabic{számláló}`, `\Roman{számláló}`, `\roman{számláló}`,
`\Alph{számláló}`, `\alph{számláló}`, `\fnsymbol{számláló}`

–érték átadás:

`\value{számláló}` használható új számláló megadásánál

`\theszámláló` aktuális számlálóérték kiírása

2.6. Feltételes utasítások

–**ifthen csomag:**

`\ifthenelse{felt}{akkor}{egyébként}`, ahol a feltétel logikai kifejezés:

`\or`, `\and` és `\not` parancsok, elemi ítéletek:

`\equal{karakterlánc}{karakterlánc}`, `\boolean{név}`,
`\isodd{szám}` stb.

`\newboolean{név}` és `\provideboolean{név}` logikai változó létrehozása, beállítása:

`\setboolean{név}{érték}`, ahol az érték lehet `true` vagy `false`

`\whiledo{felt}{végrehajtandók}` mindaddig működik, amíg igaz a feltétel

Példa:

```
\newcounter{mennyi}  
\newcommand{\MENNYI}[1]{%  
  \setcounter{mennyi}{#1}%  
  \whiledo{\value{mennyi}>0}  
    {$\circ$\addtocounter{mennyi}{-1}}}
```

Végül ennyi korong maradt: \MENNYI{7}

Végül ennyi korong maradt: ooooooooo

–**A calc csomag:** egyszerű műveletek: `\multiply`, `\divide`, `\advance`, de a calc csomag átírja

`\setcounter`, `\addtocounter`, `\setlength`, `\addtolength`

Kifejezések is megadhatóak az argumentumokban (műveleti jelek: `+`, `-`, `*`, `/`), így a `2.1mm*3` a helyes sorrend, a fordított nem!

`\real{decim}` valós szám és hányados

`\ratio{hossz1}{hossz2}` előállítás

Példa:

```
\newcounter{Ora}\newcounter{Perc}
\newcommand{\Ido}[1]{%
  \setcounter{Ora}{#1/60}%
  \setcounter{Perc}{#1-\value{Ora}*60}%
  \theOra\ óra \thePerc\ perc}
```

A percek száma `\the\time`. Így a könyvnek ezt a fejezetét épp `\Ido{\time}`kor fordította le a `\LaTeX`.

A percek száma 782. Így a könyvnek ezt a fejezetét épp 13 óra 2 perckor fordította le a `LATEX`.

2.7. Makrók és primitívek (még egy kicsit...)

–**egy kis történelem:**...Knuth és a L^AT_EX

–**definíció kiíratás:** `\show\parancs` a log állományban kiíratja a parancs definícióját, `\showváltozó` hasonlóan csak változóra

`\meaning token`, szedendő szövegbe illeszt, pl.:

`\meaning\theslide` eredmény: `macro:->\arabic {slide}`

`\fontname font` megadja a belső fonthoz tartozó külső állomány nevét, pl.:

`\fontname\font` eredmény: `ecrm1095`

`\message{üzenet a terminálra}` a log állományba és a (log)terminálra ír

–vezérlési utasítások:

`\ifcase` szám 0. eset\or 1. eset\or ...`\fi` elágazás

`\loop` első-teendő\if feltétel-tipus-teendő-ha-igaz

`\repeat` addig ismétlődik, amíg a feltétel igaz

`\endinput` befejezi egy kurrens állomány olvasását

3. Formai elemek

3.1. Betűk, szavak

–**betűméret:** a korábban megismert méretekhez újabb parancsok: `\larger[n]` és `\smaller[n]` $\mathcal{A}\mathcal{M}\mathcal{S}$ -L^AT_EX-beliek

–**betűnagyság:**

`\MakeUppercase{szöveg},`

`\MakeLowercase{szöveg},`

`\MakeTextUppercase{szöveg}` (`textcase` csomag),

`\MakeTextLowercase{szöveg}` (`textcase` csomag),

`\NoCaseChange{szöveg}` (`textcase`) parancsok

Példa:

```
\MakeUppercase{Az $f$ folytonossága.}
```

```
\MakeTextUppercase{Az $f$ folytonossága.},
```

```
\MakeTextUppercase{a \NoCaseChange{\LaTeX} ereje.},
```

AZ *F* FOLYTONOSSÁGA.

AZ *f* FOLYTONOSSÁGA.

A L^AT_EX EREJE.

–**betűritkítás:** a betűk közötti távolság növelése

`\letterspace to h\parancs{szöveg}`, ahol a `\parancs` által megadott méret `h`-szorosára változik a szöveg szélessége, `\parancs` lehet `\naturalwidth`, `\linewidth`, `\naturaldepth`, `\naturalheight`, érdemes `\mbox`-ba tenni! (`letterspace` csomag)

Példa:

```
\mbox{\letterspace to 1.5\naturalwidth{Kinn tágas,}}
```

```
\mbox{\letterspace to 0.75\naturalwidth{benn szoros!}}
```

K i n n t á g a s , bennszoros!

–**aláhúzás:** néhány érdekes példa (`ulem` csomag):

Példa:

<code>\underline{szöveg}</code>	\Rightarrow	<u>szöveg</u>
<code>\uline{szöveg}</code>	\Rightarrow	<u>szöveg</u>
<code>\uuline{szöveg}</code>	\Rightarrow	<u><u>szöveg</u></u>
<code>\sout{szöveg}</code>	\Rightarrow	szöveg
<code>\xout{szöveg}</code>	\Rightarrow	szöveg

–**parancsvégi szóköz:** például: ~~szöveg~~LaTeX utáni szóköz miatt `{}`-t kell kitenni, ez elkerülhető! (`xspace` csomag)

–**verbatim:** szó szerinti kiíratás, pl.:

```
\verb/I $\heartsuit$ \LaTeX!/
```

```
I $\heartsuit$ \LaTeX!
```

```
I ♥ LATEX!
```

3.2. Bekezdések, sorok

–**behúzás:** `\indent` és `\noindent` használható, a bekezdés behúzás mértéke:

`\setlength{\parindent}{dimen}`

–**bekezdések kizárása:**

`\begin{flushleft}...`, `\begin{flushright}...` ,

`\begin{center}...`,

érzékenység beállítható("Overfull..." hibaüzenetek):

`\fussy` (szigorú: szóközök méretében),

`\begin{fussypar}` (szigorú),

`\sloopy` (engedékeny: szóközök méretében),

`\begin{sloopypar}` (engedékeny)

–különleges formájú bekezdések:

`\shapepar{\forma}bekezdés` (shapepar csomag), ahol `\forma` adja bekezdés alakját, pl.:

`\heartpar{A a Á á B b ...}`

A a Á á B b C c
 Cs cs D d Dz dz Dzs dzs E e
 É é F f G g Gy gy H h I i Í í J j
 K k L l Ly ly M m N n Ny ny O
 o Ó ó Ö ö Ő ő P p Q q R r S
 s Sz sz T t Ty ty U u Ú u
 Ü ü Ű ű V v Z z Zs
 zs X x Y y
 ♥


–Díszes iniciálék: (yfonts)

ygoth–Gutenberg-Biblia betűtípusa

34. Sie antworteten und sprachen zu ihm: Du bist ganz in Sünden geboren und lehrest uns? Und stießen ihn hinaus. 35. Es kam vor Iesum, daß sie ihn aufgestoßen hatten.

yswab–XVIII. századi Schwabacher típus

32. Von der Welt an ist's nicht erhöret, daß jemand einem gebornen Blinden die Augen aufgetan habe. 33. Wäre dieser nicht von Gott, er könnte nichts tun.

 nd da er ihn fand, sprach er zu ihm: Glaubest du an den Sohn Gottes? 36. Er antwortete und sprach: Herr, welcher ist's ... Du hast ihn gesehen, und der mit dir redet, der ist's. 38. Er aber sprach: Herr, ich glaube; und betete ihn an.

3.3. Dobozok

–**tipusok:** LR-doboz(left-to-right), bekezdésdoboz, vonaldoboz

LR-dobozok:

`\mbox{szöveg}, \fbox{szöveg},`

`\makebox[szélesség] [poz] {szöveg},`

`\framebox[szélesség] [poz] {szöveg},`

`\raisebox{emelés} [magasság] [mélység] {szöveg}`

bekezdésdobozok:

`\parbox [poz] [magasság] [bpoz] {szélesség}{szöveg}`

`\begin{minipage} [poz] [magasság] [bpoz] {szélesség}{szöveg}`

vonaldobozok:

`\rulebox[emelés]{magasság}{szélesség}`

4. A dokumentum kinézete

4.1. Az oldal kinézetének terve

–**méretek:**

`\textheight`: szedéstükör magassága

`\textwidth`: szedéstükör szélessége

`\columnsep`: hasábok közötti távolság kéthasábos szedés esetén

`\columnwidth`: hasábszélesség (L^AT_EX kiszámolja)

`\linewidth`: sorszélesség

`\evensidemargin`: kétoldalas szedés esetén a bal margó mérete a páros oldalon

`\oddsidemargin`: bal margó a páratlan oldalon kétoldalas esetben, mindkét oldalon egyoldalas esetben

`\footskip`: az utolsó sor alapvonala és a lábjegyzet első sorának alapvonala közti távolság

`\headheight`: fejléc magassága

`\headsep`: fejléc és oldal teste közti távolság

`\topmargin`: a fejléc feletti extra térköz

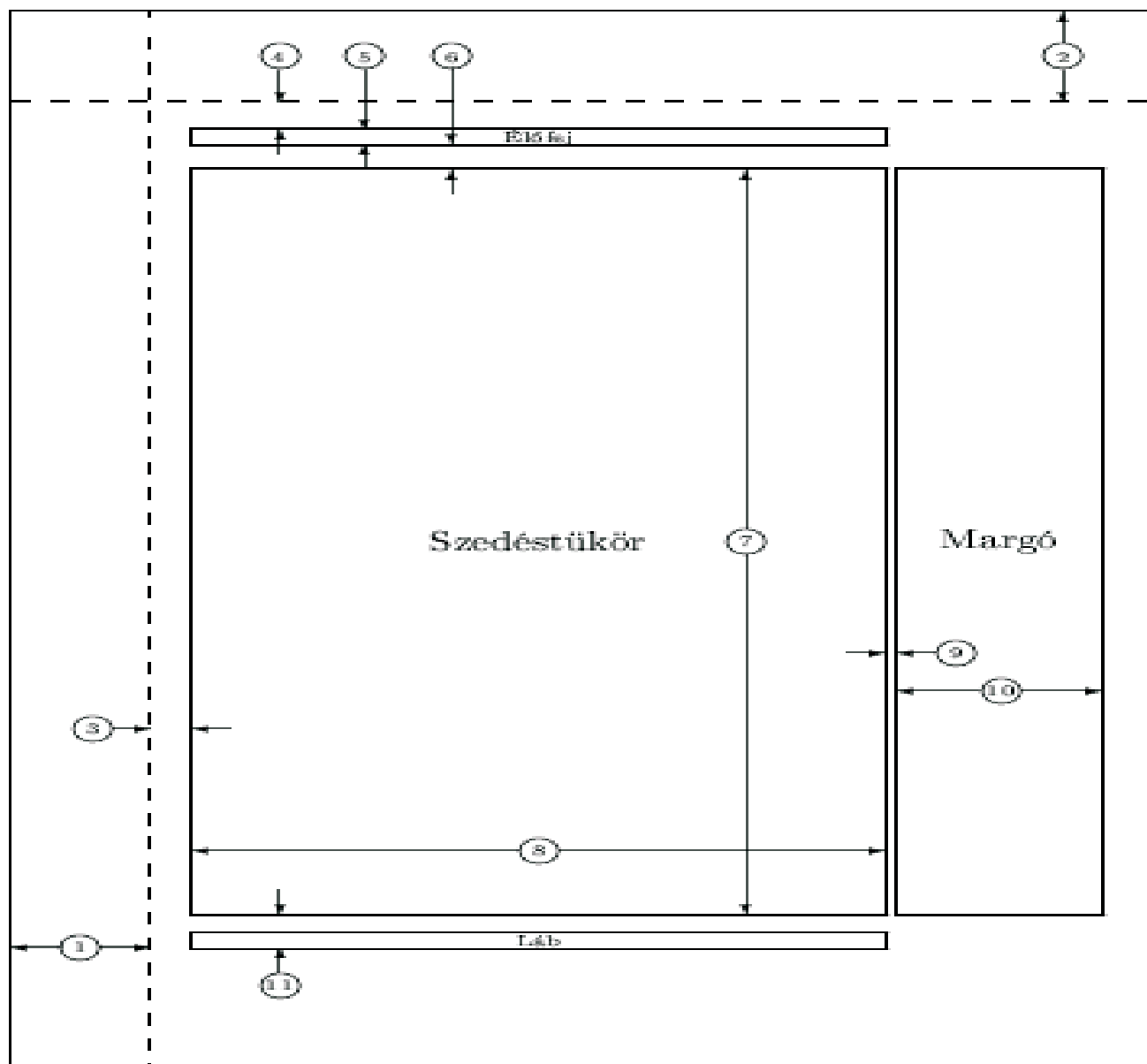
`\marginparpush`: minimális hely egymást követő széljegyzet között

`\marginparsep`: széljegyzet és a szöveg teste közti távolság

`\marginparwidth`: a széljegyzet szélessége

`\hoffset`, `\voffset`: a papír bal felső sarkának a pozícionálása
(alapértelmezés: 1-1 in)

`\paperwidth`, `\paperheight`: papír méreteinek megváltoztatása



- | | | | |
|----|------------------------|----|-------------------------------|
| 1 | egy hüvelyk + \hoffset | 2 | egy hüvelyk + \voffset |
| 3 | \oddsidemargin = 22pt | 4 | \topmargin = 22pt |
| 5 | \headheight = 13pt | 6 | \headsep = 19pt |
| 7 | \textheight = 595pt | 8 | \textwidth = 360pt |
| 9 | \marginparsep = 7pt | 10 | \marginparwidth = 106pt |
| 11 | \footskip = 27pt | | \marginparpush = 5pt (rejtve) |
| | \hoffset = 0pt | | \voffset = 0pt |
| | \paperwidth = 597pt | | \paperheight = 845pt |

4.2. Az oldal stílusa

–pagestyle:

`\pagestyle{stílus}` és `\thispagestyle{stílus}`, ahol a stílus lehet:

empty: nincs fejléc és lábléc

plain: nincs fejléc, oldalszám a lábjegyzet közepén

headings: lábléc üres, fejlécben az oldalszám, a fejezet, a szakasz vagy az alszakasz címe szerepel

myheadings: lábléc üres, fejlécben levő információt a felhasználó határozza meg (belső parancsokkal: `\@oddfoot`, `\@evenfoot`, `\@oddhead`, `\@evenhead`)

5. Fólia készítés és a texpower csomag

5.1. A slides osztály

`\begin{slide}`: minden fóliaoldalt önálló környezetbe kell tenni, a fóliák átfedhetőek: `\begin{overlay}`, `\visible` és `\invisible` parancsok, a fóliákhoz „súgó” készíthető: `\begin{notes}` segítségével

5.2. A powersem

Néhány példa a `\pause\pause` utasításra

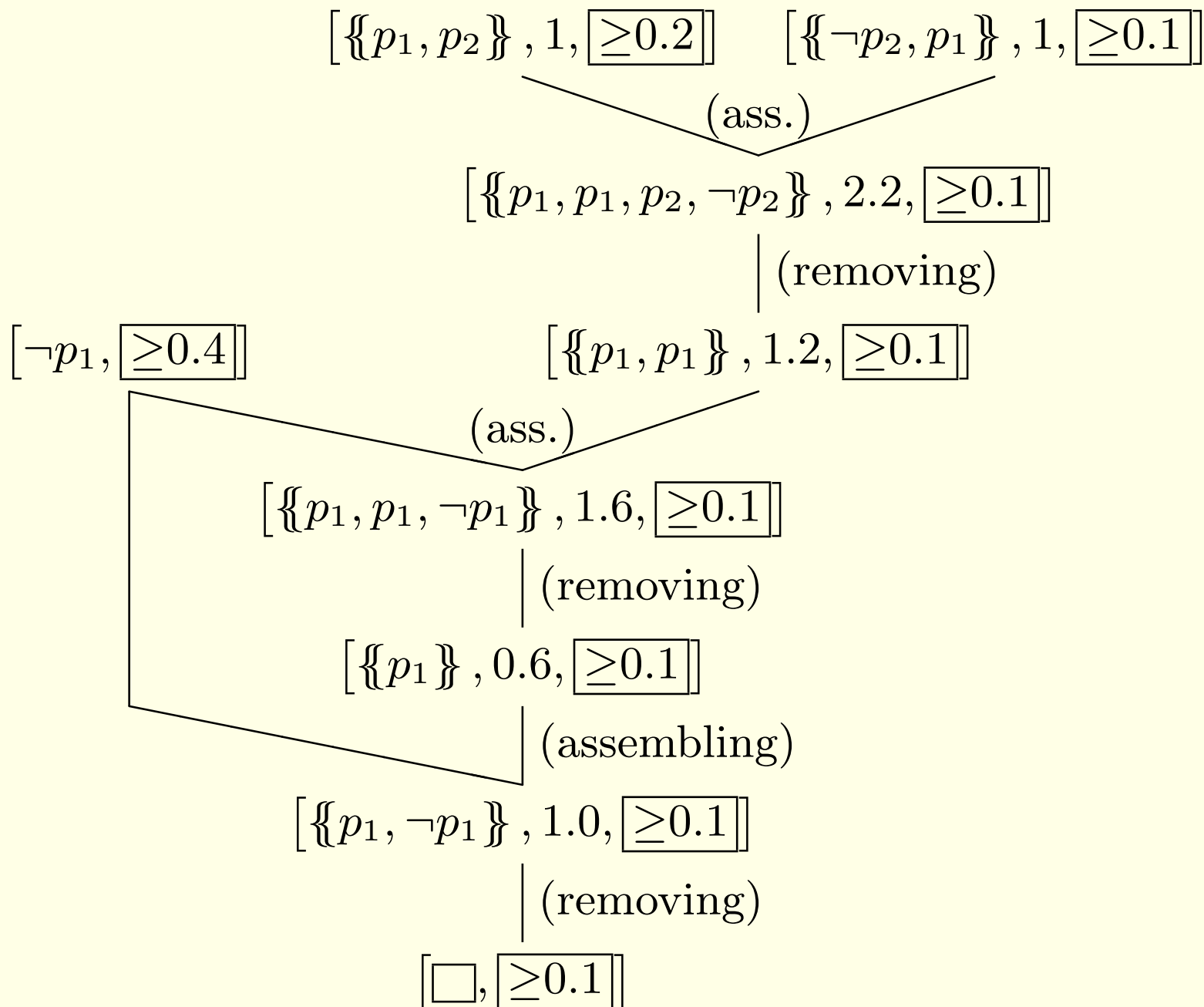
a

b

c

- foo
- bar
- baz

\stepwise példa: Egy kép



\stepwise példa: Táblázat

Táblázatot	készíthetünk	sorról sorra
vagy	celláról	cellára
vagy	akár	így is.
De	vigyázzunk a	horizontális növekedésre!

\stepwise példa: Egyenlet igazítása

$$\min \left(\max \left(\begin{array}{c} \min (F'(x), \min (F_1(x), G_1(y))), \\ \vdots \\ \min (F'(x), \min (F_n(x), G_n(y))) \end{array} \right), \min (G_i(y), H_i(z)) \right) \quad (6)$$

$$= \max \left(\begin{array}{c} \min (\min (F'(x), \min (F_1(x), G_1(y))), \min (G_i(y), H_i(z))), \\ \vdots \\ \min (\min (F'(x), \min (F_n(x), G_n(y))), \min (G_i(y), H_i(z))) \end{array} \right) \quad (7)$$

$$= \max \left(\begin{array}{c} \min (\min (F'(x), \min (F_1(x), \min (G_1(y), G_i(y)))), H_i(z)), \\ \vdots \\ \min (\min (F'(x), \min (F_n(x), \min (G_n(y), G_i(y)))) , H_i(z)) \end{array} \right) \quad (8)$$

$$= \min \left(F'(x), \min \left(\max \left(\begin{array}{c} \min (F_1(x), \min (G_1(y), G_i(y))), \\ \vdots \\ \min (F_n(x), \min (G_n(y), G_i(y))) \end{array} \right), H_i(z) \right) \right) \quad (9)$$

\stepwise példa: Bekezdésen belül

Tudunk készíteni egy
önmagát kitöltő szöveget,
amely kitölthető
bármilyen sorrendben!

Átléphetünk egy bekezdést
a szövegben sortörések
nélkül!

\stepwise példa: Visszafelé írás

Legyen most a visszafelé írás is
lehetséges!



\stepwise példa: Szöveg kiemelés

Ahelyett, hogy elemeket pusztán csak megjelenítenénk, úgy is megcsinálhatjuk, hogy a háttérből tűnjenek elő, fokozatosan

Ahelyett, hogy részletesen bemutatnánk, csak átnézünk néhány elemet, kiemelve ezeket:

» Elem 1

» Elem 2

» Elem 3

Egy bekezdésen belül ki tudunk emelni szöveget úgy, hogy nem használunk sortörést.

\stepwise példa: Bolondozgatás Ez a kis trükk megenged néhány igazán furcsa felhasználást is...

Oszthatósági demonstráció:

1

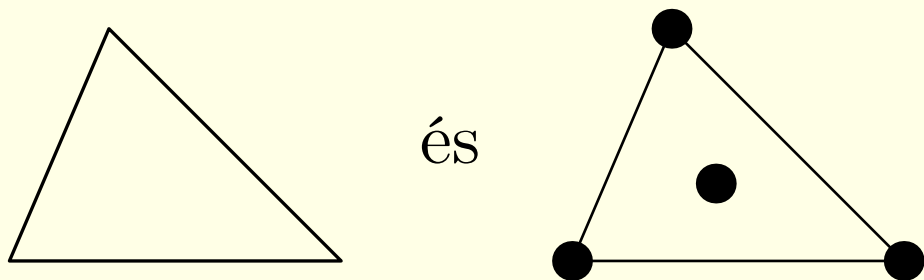
6. A METAPOST program

Egy példa (elso.mp):

```
pair A; A:=(0,0); pair B; B:=(50,0);  
pair C; C:=(15,35);  
beginfig(1)  
draw A--B--C--cycle;           % az oldalak  
endfig;  
  
beginfig(3) pickup pencircle scaled 6;  
draw A; draw B; draw C;       % a csúcsok  
draw (A+B+C)/3;               % a súlypont  
pickup pencircle scaled .4;    % alapérték  
draw A--B--C--cycle;          % az oldalak  
endfig;  
end
```


`\vcenter{\includegraphicP{elso.1}}\%`

`\quad és \quad \vcenter{\includegraphicP{elso.3}}\%`



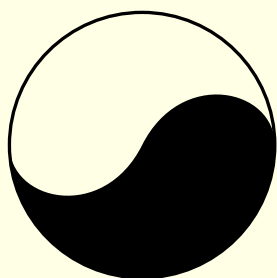
```
beginfig(1) u:=20;
```

```
path p; p:=(-u,0)..(0,-u)..(u,0);
```

```
fill p{up}..(0,0){-1,-2}..{up}cycle;
```

```
draw p..(0,u)..cycle;
```

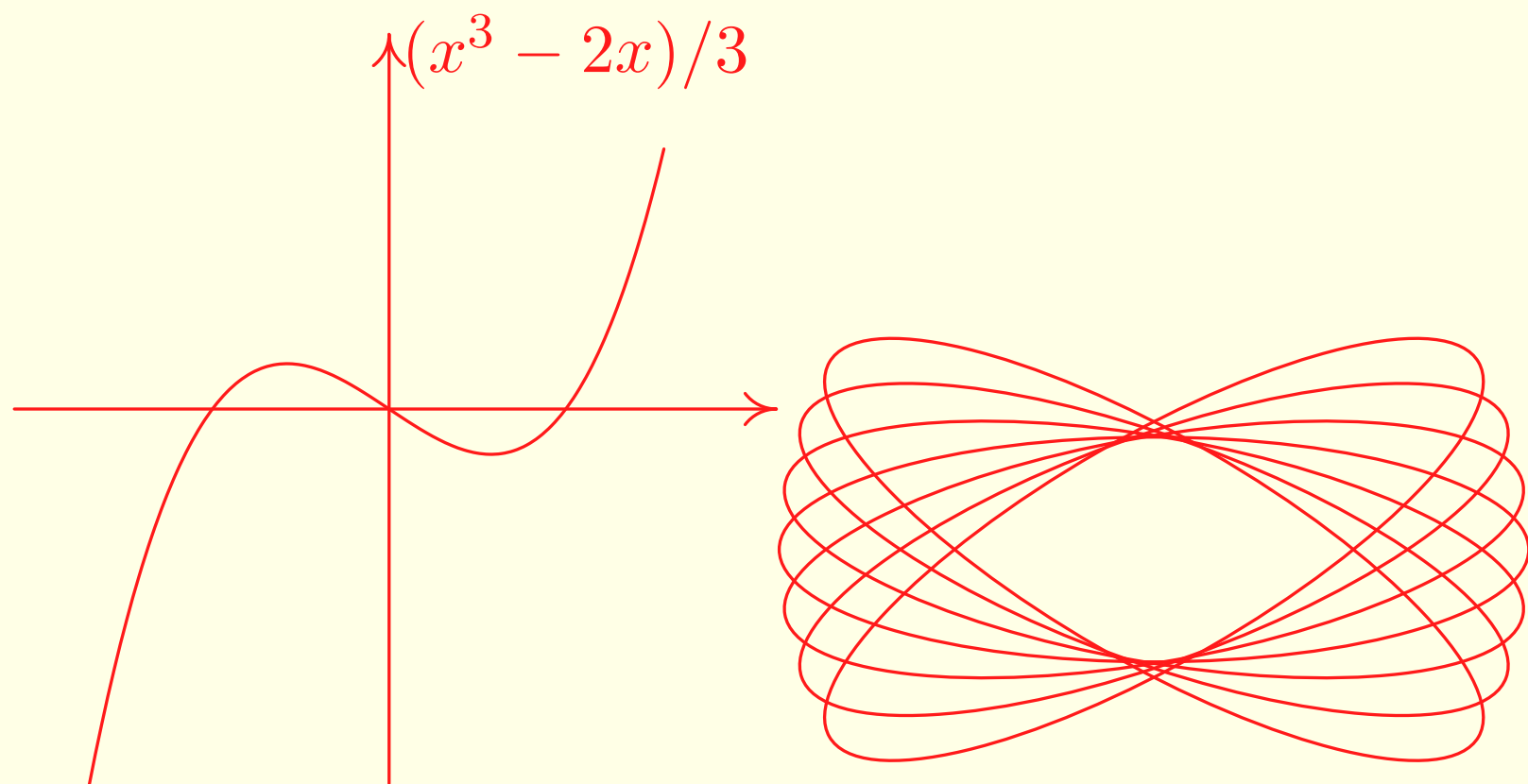
```
endfig; end
```



Az mfpic csomag

Ez már mindent tud! Metapostra épül, .mp állományt állít elő
(3 lépcsős fordítás: T_EX–mpost–T_EX)!

```
\opengraphsfile{fo}
\begin{mfpic}[20]{-3}{3.1}{-3}{3}
  \axes \function{-2.4,2.2,0.1}{((x**3)-2x)/3}
  \tlabel(0.1,2.7){$(x^3-2x)/3$}
\end{mfpic}\bigskip\newcounter{FA}
\begin{mfpic}[1.5]{-40}{40}{-25}{25}
\setcounter{FA}{-30}
\whiledo{\theFA<31}{%
  \ellipse[\theFA]{(0,0),40,12}
  \addtocounter{FA}{10}}
\end{mfpic}\closegraphsfile
```



VÉGE!!!!!! :)