

L^AT_EX informatikus hallgatók részére*

Szeidl György
Miskolci Egyetem

2002. május 17.

Tartalomjegyzék

1. Amit feltétlenül tudni kell	2
1.1. Fájlok	2
1.2. A L ^A T _E X parancsai	2
1.3. A L ^A T _E X fájl szerkezete	2
1.4. Hosszú dokumentumok	3
2. Szövegformázás	4
2.1. Sor és oldaltörések	4
2.1.1. Sortörések	4
2.1.2. Oldaltörések	5
2.2. Jelek és különleges karakterek	5
2.2.1. Idézőjelek	5
2.2.2. Kötőjelek, gondolatjel	5
2.2.3. Ligatúrák	5
2.2.4. Ékezetes betűk szedése	6
2.3. Üres helyek szavak közt	6
2.4. Bekezdések	6
2.5. Címsorok fejezetek és szakaszok	6
2.6. Alszakasz	7
2.6.1. Alalszakasz	7
2.7. Szöveg egy sorban tartása, egysoros másnéven LR dobozok	7
2.8. LR dobozok emelése és süllyesztése	8
2.9. Bekezdés dobozok	8
2.10. Dobozok mentése	9
2.11. Környezetek	9
2.11.1. Környezetek létrehozása	9
2.11.2. Listák, felsorolások	10
2.12. Balra és jobbra igazított szöveg	12
2.13. Idézetek és versek	12
2.14. Nyomtatás szó szerint	13
2.15. Tételszerű környezetek	13
2.16. Táblázatok szedése	13
2.17. Úsztatott objektumok, táblázatok	15
2.18. L ^A T _E X parancsok és környezetek létrehozása	16
2.19. Környezetek	17

*Köszönetnyilvánítás

3. Képletek szedése	17
3.1. Matematikai mód	17
3.2. A képletek elemei	18
3.2.1. Görög betűk	18
3.2.2. Indexek	18
3.2.3. Gyökjelek	19
3.2.4. Aláhúzás, föléhúzás, kapcsos zárójelek	19
3.2.5. Matematikai ékezetek	19
3.2.6. Két szélesebb matematikai ékezet	20
3.2.7. Függvénynevek	20
3.2.8. Törtek és binomiális együtthatók	20
3.2.9. Összegek, integrálok és más szimbólumok	21
3.2.10. Zárójelek, határolójelek	22
3.2.11. Kaligrafikus betűk	23
3.2.12. Helykihagyások matematikai módban	23
3.3. Többsoros képletrészek és képletek	23
3.3.1. Mátrixok	23
3.3.2. Többsoros egyenletek	24
3.3.3. Betűméret matematikai módban	24
3.4. Keretezett egyenletek kiemelt matematikai módban	25
3.5. Kövér betűk, szimbólumok	25
3.6. Az amsmath által nyújtott matematikai környezetek	26
4. További lehetőségek	27
4.1. Betűtípusok	27
4.2. Betűtnagyság	27
5. Grafika	28
5.1. Ábrák elhelyezése	28
5.1.1. WMF ábrák	31
5.1.2. GIF ábrák	32
6. Irodalomjegyzék	33
6.1. A thebibliography környezet	33
6.2. Bibliográfiai adatbázisok és használatuk	33
6.2.1. A BIBTEX program	33
6.2.2. A bibliográfiai adatbázis szerkezete	34
6.2.3. A bibliográfiai elemek lehetséges típusai	35
6.2.4. A mezők leírása	36
7. Vegyes dolgok	37
7.1. Új dokumentumosztály vagy stílus, közös néven csomag, hozzáadása	37

Kivonat

A jelen dokumentum az 2001/2002 tanév tavaszi félévében tartott előadások összefoglalója. Az idő megszabta korlátok miatt csak a legfontosabb alapfogalmak és a hallgatók külön érdeklődésére számot tartó egyes anyagrészek – pl. grafika elhelyezése – találhatók meg ebben a dokumentumban. Mivel az egyik cél egy személyi számítógépeken futtatható változat bemutatása volt különös hangsúlyt kaptak a bemutatott anyagban a Win99xx operációs rendszer alatt ingyenes MIKTEX 2.0 és 2.1 disztribúció sajátosságai.

1. Amit feltétlenül tudni kell

1.1. Fájlok

A L^AT_EX által használt leggyakoribb fájlok

StandardLatArticle01.tex – ez az a fájl amit szerkesztünk.

StandardLatArticle01.dvi – eszközfüggetlen fájl, fordítás után jön létre és pl. a MiKTEX Yap previwer-ével tekinthető meg. A dokumentum végleges alakját adja.

StandardLatArticle01.log – un. naplófájl hibaüzenetekkel.

1.2. A L^AT_EX parancsai

A L^AT_EX parancs általános szerkezete:

```
\parancsnév[kapcsolók]{kötelező paraméter}
```

vagy

```
\parancsnév[kapcsolók]{kötelező paraméter 1}{kötelező paraméter 2}
```

Például a `\setlength{\textwidth}{160mm}` parancs a szövegtükör `\textwidth` szélességét 160 mm-re állítja.

1.3. A L^AT_EX fájl szerkezete

A lentiek jól mutatják a fájl szerkezetét. Ez a fájl is ilyen szerkezetű. Felhívjuk a figyelmet arra hogy a % jellel kezdődő szövegrész – sorok illetve egy soron belül a % jel után álló szöveg – kommentnek számít, vagyis nincs hatással a dokumentum képre.

```
\documentclass[Kapcsolók]{A dokumentum osztály}%
% Kapcsolók
% -- Betű mérete: 10pt (alapértelmezés), 11pt, 12pt
% -- Papír mérete: letterpaper (alapértelmezés), a4paper, a5paper,
% b5paper, legalpaper, executivepaper
% -- Papír orientációja:
% portrait (álló, alapértelmezés)
% landscape (fekvő)
% -- Nyomtatási kép:
% oneseide (balmargó ugyanaz, alapértelmezés),
% twoside
% -- Minőség: final(alapértelmezés), draft
% -- Címloldal: notitlepage, titlepage (alapértelmezés)
% -- Hasábolás: onecolumn(egyhasábos szedés az alapértelmezés)
% twocolumn(kéthasábos szedés)
% -- Egyenletszámolás:
```

```

%                (egyenletszám jobboldalon az alapértelmezés)
%                leqno (egyenletszám jobboldalon)
%      -- Különálló egyenlet elhelyezése:
%                (középre helyezett az alapértelmezés)
%                fleqn (a bal margótól a LaTeX megszabta
%                távolságban kezdődik az egyenletszám)
%      -- Bibliográfiai stílus:
%                openbib (úgy kezeli a bibliográfiai egységek
%                fromátumát, hogy az egyes szegmensek külön
%                sorban kezdődnek)
%                Az alapértelmezés szerint ez nem lehetséges.
%
% Standard dokumentumosztályok
% article(cikk), book(könyv), report(jelentés), letter(levél), slide(fólia)
%
% Ez a dokumentum a
%
% \documentclass[a4paper,11pt]{article}
%
% paranccsal készült. A fenti parancs és a
% dokumentum kezdetét adó
%
% \begin{document}
%
% közötti rész a preambulum.
%
% Ha a magyar ékezetes betűket helyesen akarjuk írni, akkor helyezzük el
% a lenti három parancsot a preambulumban:
%
% \usepackage[USenglish,magyar]{babel}
% \usepackage{t1enc}
% \usepackage[latin2]{inputenc}
%
% és a \selectlanguage{magyar} parancsot a \begin{document} parancs után:
% \begin{document}
% \selectlanguage{magyar}
%
% Itt kezdődik a dokumentum szövege. (Valójában a \begin{document} és \end{document}
% közötti szöveg alkotja a szerkesztett dokumentumot. Szokás ezt a részt a dokumentum
% fájl törzsének is nevezni.
%
% \end{document}

```

1.4. Hosszú dokumentumok

```

\documentclass{book}
\usepackage{...}
%
% etc.
%
\includeonly{Fejezet2,Fejezet3}

```

```

\begin{document}
\frontmatter
%
% Előszó, Tartalomjegyzék
%
\mainmatter
\include{Fejezet1}
\include{Fejezet2}
\include{Fejezet3}
\include{Fejezet4}
%
%etc.
%
\include{Fejezet12}
%
\backmatter
% Irodalomjegyzék
% Tárgymutató
% Névmutató
\end{document}

```

Az `\include{Fájlnev}` parancs csak a dokumentum törzsében használható a preambulumban nem. Ha nincs beírva kiterjesztés akkor azt `tex`-nek tekinti a \LaTeX . A parancs kiadásának helyén új oldal kezdődik. Csak egyszeres mélységben használható.

Az `\input{Fájlnev}` mind a szövegtörzsben, mind pedig a preambulumban is használható. Nincs oldaltörés a parancs kiadásának helyén. Ha nincs kiterjesztés megadva a \LaTeX most is `tex` fájlt tételez fel.

Fordításkor, ha van `includonly` parancs a preambulumban, a \LaTeX csak a parancs argumentumában felsorolt fájlokból készíti `dvi` fájlt

2. Szövegformázás

2.1. Sor és oldaltörések

2.1.1. Sortörések

A dokumentum minta a standard \LaTeX cikk osztályra. Figyeljük meg, hogy az első bekezdés nem kezdődik beljebb.

Mindjárt eltörjük a sort a `\[5mm]` parancs segítségével. A parancs kiadása után a kurrens sor balra igazított. Ezt a sort 5 mm helyköz követi, és az új sor identáció nélküli. Ha közben laptörésre kerül sor, pl. 3 mm-el az eltört sor alatt, akkor elvész a további 2 mm helyköz, és az új sor lap tetején identáció nélküli.

A `\newline` parncs azonos a `\` parancssal.

Az itt következő szöveg demonstrálja a fentieket.

Hétköznapi megfigyelés, hogy ugyanazon szilárd test alakváltozásainak mértéke függ a testet terhelő erőrendszertől. Minél nagyobb a terhelés annál nagyobb alakváltozások figyelhetők meg. A statikailag határozott rudak igénybevételeinek számításánál azt is láttuk, hogy a feszültségi eredők, következésképp a belső erőrendszer (sortörés!)

megoszlása a rúdkeresztmetszeteken függ a terheléstől. Ez természetesen más, nem rudalakú

testeknél is így van. Következik tehát hogy a szilárd test terhelés hatására kialakuló belső erőrendszere és alakváltozási állapota egymással kapcsolatban van.

Sortörés érhető el a `\linebreak[n]` $n=1,2,3,4$ parancs kiadásával is. Az n számláló a parancs erősségét jelzi. Ha nincs szám megadva, akkor 4 az n . A különbség az előző parancshoz képest abban jelentkezik, hogy az eltört sor sorkiegyenlített marad („teljes” sor marad)

A `\nolinebreak[n]` parancs a sortörést gátolja meg ott, ahol kiadjuk.

Bekezdés után kis, közepes és viszonylag több hely hagyható ki a

`\smallskip` `\medskip` és `\bigskip`

parancsok segítségével.

2.1.2. Oldaltörések

Oldaltörés érhető el a `\newpage` parancs segítségével. Hatásai:

1. Ha bekezdésben adjuk ki, akkor a parancs kiadása helyén új oldalt kezd. A kurrens sor parancs előtti része balra igazított marad.
2. Ha bekezdés végén adjuk ki, akkor nincs sortörés.
3. Az oldal parancskiadás helyétől jobbra és alatta lévő része üres hellyel töltődik fel.
4. A kurrens sor parancskiadás utáni része az új oldal tetején identációval jelenik meg.

Oldaltörés érhető el a `\pagebreak{n}` parancs segítségével is. Hatásai:

1. Ha bekezdésben adjuk ki, akkor a parancs kiadásának sora azon a lapon marad, ahol a parancsot kiadtuk. A sor természetesen sorkiegyenlített marad.
2. Az oldal fennmaradó része üres hellyel töltődik fel.
3. A következő sor az új oldal tetején identáció nélkül jelenik meg.

A `\nopagebreak[n]` parancs az oldaltörést gátolja meg az után a sor után, ahol kiadjuk.

2.2. Jelek és különleges karakterek

2.2.1. Idézőjelek

Angol: ‘‘English’’, azaz “English”

Angol de kicsit másképp: ‘English’, azaz ‘English’

Magyar: „Magyar idézőjel’’, azaz „Magyar idézőjel”

2.2.2. Kötőjelek, gondolatjel

Magyar kötőjel: - pl.: X-akták (az angol is ugyanez)

Magyar gondolatjel: -- pl. Lenni – vagy nem lenni.

Angol gondolatjel: --- pl. To be — or not to be. Magyar szövegben nem használjuk.

Mínusz jel: \$-\$ pl. -45.

2.2.3. Ligatúrák

Az `ff`, `fi`, `fl`, `ffi` betűkombinációk a ligatúrák.

Kifinomult, Szaffi, de nem széffúró, hanem széf{ }fúró.

Kifinomult, Szaffi, de nem széffúró, hanem széffúró.

2.2.4. Ékezetes betűk szedése

A táblázat mutatja a lehetőségeket:

<code>\'{} = ó</code>	<code>\'{} = ò</code>	<code>\^{o} = ô</code>	<code>\"{} = ö</code>	<code>\~{} = õ</code>
<code>\={o} = õ</code>	<code>\.{} = ò</code>	<code>\u{} = ü</code>	<code>\v{} = v</code>	<code>\H{} = H</code>
<code>\t{} = t</code>	<code>\c{} = c</code>	<code>\d{} = d</code>	<code>\b{} = b</code>	<code>\r{} = r</code>

2.3. Üres helyek szavak közt

Fix szóköz `_` egy fix szóközt ad. (A rövid lent elhelyezett vízszintes egy helyközt jelent.)
pl. Itt `_` minden `_` helyköz `_` ugyanaz. = Itt minden helyköz ugyanaz.

A mondatvégi pont után több kell legyen a hely, ha az utolsó betű kisbetű.

Ha nem – pl. BASIC. Következő mondat eleje stb. – akkor a `\@` parancsot kell használni:

pl. BASIC.`\@` Következő mondat eleje stb. =
= pl. BASIC. Következő mondat eleje stb.

`\franchspacing` egyforma helyközre törekszik mindenütt, a mondatvégeken is.

`\nofranchspacing` egyforma helyközre törekszik a mondaton belül, a mondatvégeken azonban több hely.

2.4. Bekezdések

A `\parskip` rugalmas távolság a két bekezdés közötti távolság. Függsz a betűmérettől.

A `\parindent` a bekezdés első sorának identációja. Ez is függ a betűmérettől.

A `\baselinestretch` azt mondja meg, hogy hányszorosa a két szomszédos sor alapvonala közti távolság az alapértelmezés szerinti távolságnak. Kezdeti értéke 1.

Az értékét megváltoztatható a

```
\renewcommand{\baselinestretch}{1.5}
```

parancs révén. A parancs csak akkor lép életbe ha megváltoztatjuk a betűnagyságot is. Ez azt jelenti, hogy a

```
\renewcommand{\baselinestretch}{1.5}\small\normalsize
```

parancsok kiadása után másfélszeres az alapsortáv:

Vegyük észre, hogy a fentiek, így az utolsó kérdés megfogalmazása is, egy hallgatólagos feltevést tartalmaz, nevezetesen hogy a feszültségi tenzor csak az alakváltozási tenzor függvénye, azaz független más mennyiségektől, így a terhelés történetétől, vagy mondjuk a hőmérsékletváltozástól. Érdeemes ezen a ponton hangsúlyozni, hogy a hallgatólagos feltevés miszerint létezik a kölcsönösen egyértelmű etc.

A `\renewcommand{\baselinestretch}{1.0}\small\normalsize` parancs visszaállítja az eredeti sortávot.

2.5. Címsorok fejezetek és szakaszok

A \LaTeX támogatja a következő részre, fejezetre, szakaszra tagoló parancsokat:

`\part` – rész; Book és report dokumentumosztályra
`\chapter` – fejezet; Book és report dokumentumosztály

<code>\section</code>	– fejezet; Book, report és article dokumentumosztály
<code>\subsection</code>	– alfejezet (alszakasz); Book, report és article dokumentumosztály
<code>\subsubsection</code>	– alalfejezet (alalszakasz); Book, report és article dokumentumosztály
<code>\paragraph</code>	– bekezdés; Book, report és article dokumentumosztály
<code>\subparagraph</code>	– albekezdés; Book, report és article dokumentumosztály
<code>\appendix</code>	– függelék; Book, report és article dokumentumosztály
A <code>\begin{document}</code>	parancsot az un. frontmatter követi.

2.6. Alszakasz

A

```
\subsection{Alszakasz\label{cimke}}
```

parancs a jelen alszakaszt hozta létre. A

```
\label{cimke}
```

parancs címkét rendel az alszakasz sorszámához illetve ahhoz az oldalhoz, ahol a

```
\subsection{Alszakasz\label{cimke}}
```

parancs kiadására sor került. Az alszakasz számra a `\ref{cimke}` parancs kiadásával, magára az oldalszámra pedig a `\pageref{cimke}` parancs révén hivatkozhatunk: ezek szerint most a 2.6 alszakaszban vagyunk, amely a 7. oldalon kezdődik. Vagy most a 2.17 szakaszban vagyunk, amely a 16-ik oldalon kezdődik.

2.6.1. Alalszakasz

Itt pedig egy alalszakasz látszik.

Ez egy paragrafus. Ez egy rövid mondat amit a paragrafusba szántunk.

Alparagrafus. És még ezt is lehet fokozni. Különösen ha akarjuk.

2.7. Szöveg egy sorban tartása, egysoros másnéven LR dobozok

Az

1. `\mbox{text}` illetve a `\makebox[szélesség][pozíció]{szöveg}`, továbbá az
2. `\fbox{text}` illetve a `\framebox[szélesség][pozíció]{szöveg}` parancsok

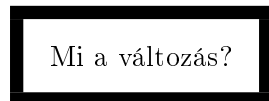
keretezetlen illetve keretezett dobozokat hoznak létre. Például

```
\fbox{Mi a változás?} = Mi a változás?
```

A szélesség a doboz szélessége. (A fenti parancsok csak „egy sorban lévő szöveget, dobozt” tudnak létrehozni.) A pozíció értéke

- ha *c*, akkor a szöveg középre igazított;
- ha *l*, akkor a szöveg balra igazított;
- ha *r*, akkor a szöveg jobbra igazított;
- ha *s*, akkor a szöveg „sorkiegyenlített” azaz kitölti egyenletesen a szélesség megszabta helyet.

A `\setlength{\fboxrule}{5pt}` `\setlength{\fboxsep}{10pt}` parancsok változást eredményeznek. Az első a vonalvastagságot, a második a szöveg és keret közötti távolságot állítja.



Próbálja kitalálni hogyan szedtük az alábbi szövegrészeket:

centered vagy §

2.8. LR dobozok emelése és süllyesztése

A `\raisbox{emelés mértéke}[magasság][mélység]{szöveg}` parancs egy `\mbox`-ot hoz létre majd azt az emelés mértékével felemeli, vagy lesüllyeszti – ha negatív az emelés – a sor alapvonalához képest.

A magasság és mélység, ha megadjuk, a doboz mérete az alapvonalhoz képest felfelé illetve lefelé. Az alábbi sor a parancs használatát illusztrálja.

```
\noindent Az alapvonal, \hrulefill\ \raisebox{1ex}{most fel picit} %
\hrulefill\ \raisebox{-1ex}{most pedig le egy picit} \hrulefill\ most %
ismét alapvonal.
```

Az alapvonal, _____ most fel egy picit _____ most pedig le egy picit _____ most ismét alapvonal.

2.9. Bekezdés dobozok

Az alábbi két parancssal un. bekezdésdobozok hozhatók létre:

```
\parbox{pozíció}{doboz szélessége}{szöveg}
\begin{minipage}{pozíció}{doboz szélessége} szöveg \end{minipage}
```

A pozíció értéke `b` (bottom) és `t` (top) lehet. A `b` paraméter megadása esetén a doboz alsó széle a sor alapvonalához igazított, ha `t` a pozíció akkor a doboz felső széle van a sor alapvonalán. Ha nincs megadva pozíciót jelző paraméter, akkor a doboz közepe van az aktuális sor alapvonalán.

Példák:

```
\noindent%
\parbox{4cm}{\sloppy Ez egy mindössze 4 cm széles parbox.
a doboz közepe a sor alapvonalán van.}\hfill aktuális sor \hfill
\parbox[t]{4cm}{\sloppy A keskeny szövegkasákokat nehéz sorkiegyenlített
módban szedni. Sok hibaüzenet kerül ui. ekkor a naplófájlba. Ezen a
{\tt\symbol{92}sloppy} parancs kiadása segíthet}
```

Ez egy mindössze 4 cm
széles parbox. a doboz
közepe a sor alapvona-
lán van.

aktuális sor

A keskeny szövegkasá-
kokat nehéz sorkiegyen-
lített módban szedni.
Sok hibaüzenet kerül
ui. ekkor a naplófájlba.
Ezen a `\sloppy` parancs
kiadása segíthet

```
\noindent\fbox{%
\begin{minipage}[b]{45mm}
```

A `\symbol{92}minipage` környezet jóval flexibilisebb mint a `\symbol{92}parbox` környezet. Ennek az az oka, hogy más, pl. listázó környezetek is elhelyezhetők benne.

Ennek a `\symbol{92}minipage`-nek az alsó széle van az alapvonalon. `\mbox{}`

```
\end{minipage}}%
```

```
\hrulefill\fbbox{%
```

```
\begin{minipage}{45mm}
```

A mellette elhelyezett `\symbol{92}minipage` környezetnek már a középvonala

található az alapvonalon, mivel nincs megadva a pozíciót kijelölő paraméter.

```
\end{minipage}}%
```

```
\hrulefill\fbbox{%
```

```
\begin{minipage}[t]{45mm}\mbox{}
```

Végezetül úgy rendeztük el a harmadik `\symbol{92}minipage` környezet, hogy ennek a felső széle legyen az alapvonalon. Ehhez csupán a `\texttt{t}` paraméter megadására volt szükség.

```
\end{minipage}}%
```

A `\minipage` környezet jóval flexibilisebb mint a `\parbox` környezet. Ennek az az oka, hogy más, pl. listázó környezetek is elhelyezhetők benne. Ennek a `\minipage`-nek az alsó széle van az alapvonalon.

A mellette elhelyezett `\minipage` környezetnek már a középvonala található az alapvonalon, mivel nincs megadva a pozíciót kijelölő paraméter.

Végezetül úgy rendeztük el a harmadik `\minipage` környezet, hogy ennek a felső széle legyen az alapvonalon. Ehhez csupán a `t` paraméter megadására volt szükség.

2.10. Dobozok mentése

Ha ugyanaz a szöveg több helyen is előfordul egy dokumentumban, akkor az eltárolható első lépésben a

```
\newsavebox{\doboznév}
```

parancs kiadásával. A doboznév csak betűket tartalmazhat és nem lehet létező L^AT_EX parancs neve.

Miután sor került a doboznév létrehozására a

```
\sbox{\doboznév}{szöveg}
```

```
\savebox{\doboznév}[szélesség][pozíció]{szöveg}
```

```
\begin{lrbox}{\doboznév} szöveg \end{lrbox}
```

parancsok elmentik a szöveget későbbi felhasználásra, ami a

```
\usebox{\doboznév}
```

parancs kiadásával történhet. Ez beszúrja a szöveget a parancskiadás helyén a dokumentumba.

2.11. Környezetek

2.11.1. Környezetek létrehozása

Egy környezet a

```
\begin{Környezet neve} Környezet szövege \end{Környezet neve}
```

paranccsal hozható létre.

2.11.2. Listák, felsorolások

Három tipikus listakörnyezet létezik:

```
\begin{enumerate} listaszöveg \end{enumerate}
\begin{itemize} listaszöveg \end{itemize}
\begin{description} listaszöveg \end{description}
```

Az alábbiak sorra veszik egyes listakörnyezeteket.

Az `enumerate` környezet:

```
\begin{enumerate}
\item Első listaelem
\item Második listaelem
\begin{enumerate}
\item Listaelem a listaelem alatt
Ennek más a stílusa. A listák négyszeres
mélységig skatulyázhatók egymásba.
\item Egy másik listaelem a listaelem alatt.
\begin{enumerate}
\item Ez a harmadik szint.
\begin{enumerate}
\item Ez pedig a negyedik.
\end{enumerate}
\end{enumerate}
\end{enumerate}
\end{enumerate}
```

És ezt kapjuk szedés után:

1. Első listaelem
2. Második listaelem
 - (a) Listaelem a listaelem alatt Ennek más a stílusa. A listák négyszeres mélységig skatulyázhatók egymásba.
 - (b) Egy másik listaelem a listaelem alatt.
 - i. Ez a harmadik szint.
 - A. Ez pedig a negyedik.

A pontozott, azaz `itemize` lista négy szintje:

```
\begin{itemize}
\item Első listaelem
\item Második listaelem
\begin{itemize}
\item Listaelem a listaelem alatt
Ennek más a stílusa. A listák négyszeres
```

```

mélységig skatulyázhatók egymásba.
\item Egy másik listaelem a listaelem alatt.
\begin{itemize}
  \item Ez a harmadik szint.
  \begin{itemize}
    \item Ez pedig a negyedik.
  \end{itemize}
\end{itemize}
\end{itemize}
\end{itemize}

```

És így néz ki szedés után:

- Első pontozott listaelem
- Második pontozott listaelem
 - Listaelem a listaelem alatt Ennek más a stílusa. A listák négyszeres mélységig skatulyázhatók egymásba.
 - Egy másik listaelem a listaelem alatt.
 - * Ez a harmadik szint.
 - Ez pedig a negyedik.

A `description` környezet is listák szedésére készül. A lista elemei előtt azonban nincs szám, hanem a listaelem néhány szava van külön is kiemelve. Részlet a hirdetményből:

```

\begin{description}
\item[Február 16.] $\rule[0.0mm]{0.0mm}{0.5mm}$\newline
Miért \LaTeX{} ? Érvek pro és kontra.\
Amit feltétlenül tudni kell.
\begin{itemize}
\item[-- ] A \LaTeX{} kézirat. (Speciális karakterek. \LaTeX{} parancsok.
Csoportosítás. A kéziratfájl szerkezete.)
\item[-- ] A dokumentum típusa. (Dokumentumosztályok. Csomagok.
Oldalstílusok. Hosszú dokumentumok.) stílus
\end{itemize}
\noindent Szövegformázás I.
\end{description}

```

vagy ami ugyanaz:

Február 16.

Miért \LaTeX ? Érvek pro és kontra.

Amit feltétlenül tudni kell.

- A \LaTeX kézirat. (Speciális karakterek. \LaTeX parancsok. Csoportosítás. A kéziratfájl szerkezete.)
- A dokumentum típusa. (Dokumentumosztályok. Csomagok. Oldalstílusok. Hosszú dokumentumok.)

Szövegformázás I.

2.12. Balra és jobbra igazított szöveg

Balra igazított szöveget a `flushleft` környezet segítségével szedhetünk:

```
\begin{flushleft}
  Balra igazított szöveg szedése.
  Ez a szöveg mindössze két sorból áll.
\end{flushleft}
```

Balra igazított szöveg szedése. Ez a szöveg mindössze két sorból áll.

Jobbra igazított szöveg esetén `flushright` a környezet neve:

```
\begin{flushright}
  Jobbra igazított szöveg szedése.
  Ez a szöveg mindössze két sorból áll.
\end{flushright}
```

Jobbra igazított szöveg szedése. Ez a szöveg mindössze két sorból áll.

2.13. Idézetek és versek

Idézett szöveg a `quote` környezet segítségével szedhető. A szöveg baloldala beljebb van és nincs identáció:

```
\begin{quote}
  Mondottam ember, hogy küzdj és
  bízva bízzál. \emph{Madách Imre}
\end{quote}
```

Mondottam ember, hogy küzdj és bízva bízzál. *Madách Imre*

A `quotation` környezet használata esetén mindkét oldal beljebb szedett és minden bekezdésnek van identációja:

```
\begin{quotation}
  Volt egyszer egy földbe vájt lyuk, ebben élt egy babó. Nem volt ez
  a lyuk rút, mocskos, nedves, teli féregmaradékkal, dohszaggal, sem száraz
  csupasz, homokos lyuk, ahol se leülni sem enni nemigen lehet; ez babólyuk
  volt ami egyértelmű a kényelemmel.

  Olyan kerek, zöldre mázolt ajtaja volt, mint a kajütblak, s annak pontosan
  a közepén fénylő rézkilincs. \emph{J.R.R. Tolkien: A babó}
\end{quotation}
```

Volt egyszer egy földbe vájt lyuk, ebben élt egy babó. Nem volt ez a lyuk rút, mocskos, nedves, teli féregmaradékkal, dohszaggal, sem száraz csupasz, homokos lyuk, ahol se leülni sem enni nemigen lehet; ez babólyuk volt ami egyértelmű a kényelemmel.

Olyan kerek, zöldre mázolt ajtaja volt, mint a kajütblak, s annak pontosan a közepén fénylő rézkilincs. *J.R.R. Tolkien: A babó*

Versszöveg, drámaszöveg a `verse` környezet segítségével szedhető. A szöveg baloldala beljebb van és nincs identáció:

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

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

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

Gyepes hanton furulyált,
Legelészett nyája
...

2.14. Nyomtatás szó szerint

Többsoros szöveg esetén a

```
\begin{verbatim} többsoros szöveg \end{verbatim}
\begin{verbatim*} többsoros szöveg \end{verbatim*}
```

környezetek használhatók. A környezetben belül kiadott L^AT_EX parancsok nem hajtódnak végre.

Egy soron belül a

```
\verb+ egy sornál rövidebb szöveg +
\verb** egy sornál rövidebb szöveg +
```

parancsok használhatók. A szöveg kezdetét és végét jelző karakterkettős, a példánkban ez a + jel, nem szerepelhet a szövegben. A csillagos változatok □ módon jelölik a helyközt.

2.15. Tételszerű környezetek

A preambulumban kiadott

```
\newtheorem{környezetnév}{sorszámozott nyomtatásban megjelenő név}
```

paranccsal hozhatók létre. Ha

```
\newtheorem{mytheorem}{tétel}
```

volt a parancs, akkor a

```
\begin{mytheorem}
```

A derékszögű háromszög két befogója négyzetének összege megegyezik az átfogó négyzetével. (Ez a tétel Pythagoras tétel néven ismert.)

```
\end{mytheorem}
```

szöveg szedésének:

1. tétel. *A derékszögű háromszög két befogója négyzetének összege megegyezik az átfogó négyzetével. (Ez a tétel Pythagoras tétel néven ismert.)*

az eredménye.

2.16. Táblázatok szedése

Táblázatok a

```
\begin{array}[pozíció]{oszlopok} táblázat sorai \end{array} -- matematikai mód
\begin{tabular}[pozíció]{oszlopok} táblázat sorai \end{tabular}
```

parancsokkal hozhatók létre.

A táblázat, hasonlóan a bekezdéskörnyezetekhez, egy „betűnek” számít. A pozíció értéke **t** és **b** lehet, a korábbi jelentéssel – **t** esetén a táblázat mint betű teteje az alapvonalon van, **b** esetén pedig a táblázat mint betű alja van az alapvonalon:

```
\noindent\hrulefill%
\begin{tabular}[t]{|c|c|}\hline
  1 & 2 \\ \hline % ez a táblázat első sora
  3 & 4 \\ \hline % ez a táblázat második sora
\end{tabular}\hrulefill%
\begin{tabular}   {|c|c|}\hline
  1 & 2 \\ \hline % nincs pozíció, a táblázat
  3 & 4 \\ \hline % függőlegesen középre igazított
\end{tabular}\hrulefill%
```


Tizedespont kezelése: az `@{c}` parancs nullára csökkenti két szomszédos oszlop távolságát, a két oszlopot pedig `c` betű választja el. Pl.:

```
\begin{tabular}{|r|r@{.}l|} \hline
  1 & 26&46 \\ \hline
  2 & 45&615 \\ \hline
  3 & 625&187 \\ \hline
\end{tabular}
```

1	26.46
2	45.615
3	625.187

Oszlopok egyesítése: a `\multicol{n}{oszlopok}{szöveg}` parancs egyesíti a parancskiadás után álló n számú oszlopot, az *oszlopok* az *oszlopok* leírására szolgáló valamelyik paraméter, amelyet megelőzhet, vagy követhet a `|` karakter, a szöveg pedig az egyesített oszlopokban jelenik meg – egy sorban több ilyen parancs is kiadható. Pl.:

```
\begin{tabular}{|c|c|c|} \hline
 \multicolumn{3}{|l|}{\textbf{Három egyesített oszlop}} \\ \hline
 \multicolumn{2}{|l|}{Két egyesített oszlop} & Harmadik oszlop \\ \hline
 Első oszlop & \multicolumn{2}{|l|}{Ismét két egyesített oszlop} \\ \hline
 123 & 241 & 12.5 \\ \hline
\end{tabular}
```

Három egyesített oszlop		
Két egyesített oszlop		Harmadik oszlop
Első oszlop	Ismét két egyesített oszlop	
123	241	12.5

2.17. Úsztatott objektumok, táblázatok

Az úsztatott elhelyezést a

```
\begin{table}[hol] \caption[rövid cím]{teljes cím} \label{címke} táblázat \end{table}
\begin{table*}[hol] \caption[rövid cím]{teljes cím} \label{címke} táblázat \end{table}
```

parancsok teszik lehetővé. A parancs csillagos változata kéthasábos szedés esetén használatos. Ekkor a táblázat kiterjed szélességben mindkét hasábra.

A hol paraméter a táblázat helyét szabja meg, értékei:

- h (here=itt): a táblázat lehetőleg ott jelenjen meg, ahol a *table* környezetet beírjuk a szövegbe;
 - t (top=lap teteje): a táblázat lehetőleg az aktuális lap tetején jelenjen meg;
 - b (bottom=lap alja): a táblázat lehetőleg az aktuális lap alján jelenjen meg;
 - p (page=külön lap): a táblázat lehetőleg az úszó objektumok részére fenntartott külön lapon jelenjen meg;
- ! enyhít itt nem részletezett korlátozásokat – részleteket illetően lásd [1, 174-175 o.], [3, 258-260 o.]

A táblázatok elhelyezkedését illetően az alábbiakat érdemes említeni:

1. Az úszó táblázat nem kerülhet laphatáron kívülre.
2. Az úszó táblázat vagy azon az oldalon jelenik meg, ahol a `table` környezetet létrehoztuk, vagy pedig az ezt az oldalt követő oldalak valamelyikén.

- Az úszó táblázatok egymáshoz viszonyított sorrendje változatlan marad bárhova is helyezze el őket egyébként a \LaTeX .
- A `[htb]` vagy `[htbp]` opciók bármilyen más sorrendben is írhatók, mivel a \LaTeX a sorrendtől függetlenül először a parancskiadás helyén próbálja elhelyezni az úszó táblázatot; ha ez nem megy akkor a lap tetejére, ha ez sem megy, akkor pedig a lap aljára kerül a táblázat. Ha az előző esetek egyike sem valósítható meg, akkor pedig külön oldalon jelenik meg az úszó táblázat.

A `\caption[rövid cím]{teljes cím}` parancs kiadásának helyén automatikus sorszámmal ellátott feliratot kap a táblázat. A rövid cím, ha megadjuk, a táblázatok jegyzékében megjelenő cím. A paranccsal megadott címke révén hivatkozni lehet az úszó táblázatra. Az alábbi példa az előző szakasz végén lévő táblázatot úszó környezetbe helyezi:

```
\begin{table}[htb]
\caption{Példa oszlopok egyesítésére}\label{Oszlopok}\vspace{1.5mm}
\centering
\begin{tabular}{|c|c|c|} \hline
\multicolumn{3}{|c|}{\textbf{Három egyesített oszlop}} \\ \hline
\multicolumn{2}{|c|}{Két egyesített oszlop} & Harmadik oszlop \\ \hline
Első oszlop & \multicolumn{2}{|c|}{Ismét két egyesített oszlop} \\ \hline
123 & 241 & 12.5 \\ \hline
\end{tabular}
\end{table}
\noindent A fenti \ref{Oszlopok}. táblázat oszlopok egyesítését szemlélteti.
```

1. táblázat. Példa oszlopok egyesítésére

Három egyesített oszlop		
Két egyesített oszlop		Harmadik oszlop
Első oszlop	Ismét két egyesített oszlop	
123	241	12.5

A fenti 1. táblázat oszlopok egyesítését szemlélteti.

2.18. \LaTeX parancsok és környezetek létrehozása

\LaTeX parancsok a preambulumban elhelyezett parancsdefinícióval hozhatók létre. Ennek a következő a szintaxisa:

```
\newcommand{\parancsnév}[paraméterek száma]{a parancs definíciója} illetve
\renewcommand{\parancsnév}[paraméterek száma]{a parancs definíciója}.
```

A (3.2.9) szakasz táblázatának szedése során például a

```
\newcommand{\SymbName}[1] % A parancs neve és paramétereinek száma.
{\begin{minipage}{6mm} % A parancs definíciója. Figyeljük meg, hogy a paraméterre
  \[ #1 \] % a #1 módon hivatkozunk. Több paraméter esetén is hason-
\end{minipage}} % ló módon járunk el: #3 a harmadik paraméterre hivatkozik.
```

parancsot használtuk fel. Ez az egyparaméteres parancs hat mm széles dobozt hoz létre és ebben a dobozban szedjük egyenletként a paraméreként átvett szimbólumot.

2.19. Környezetek

```
\newcounter{Rem}
\newenvironment{Megjegyzes}
{\begin{sloppypar}\noindent\stepcounter{Rem}\arabic{Rem}. \textsc{Megjegyzés}:
\begin{quote}\small\itshape}%
{\end{quote}\end{sloppypar}}
```

1. MEGJEGYZÉS:

Ez egy rövid apróbetűs megjegyzés lesz.

3. Képletek szedése

3.1. Matematikai mód

Mivel a matematikai mód eszköztárát jelentősen megnövelik az `amsmath`, `amsfonts` és `amssymb` csomagok, feltételezzük, hogy ezek is be vannak töltve – ellenőrizze a jelen dokumentum preambulumbát! A sorvégi `\in` csomag szöveg arra utal, hogy az adott parancs része az említett csomagoknak.

Sorközi matematikai módról beszélünk ha a képlet folyamatosan szedett szövegben jelenik meg. Sorközi matematikai módba a

```
\begin{math} képlet \end{math}, \left( képlet \right) és a $ képlet $
```

parancsok egyikével lehet átváltani. Rövidsége és nem törekeny volta miatt az utolsót érdemes használni.

Kiemelt matematikai módba egysoros egyenlet esetén a

```
[\ képlet szövege \]
\begin{displaymath} képlet szövege \end{displaymath}
\begin{equation*} képlet szövege \end{equation*} \in amsmath
```

illetve a

```
\begin{equation} képlet szövege \label{címke}\end{equation} \in amsmath, LATEX
```

parancsokkal lehet áttérni.

Az első három esetben nincs automatikus képletszám és ezek a parancsok egyenértékűek is egymással. A negyedik esetben automatikus a képletszámozás és a képletszámra a címke segítségével hivatkozni is lehet a `\ref{címke}` parancssal.

Egysoros képlet például a

```
\begin{equation}
y^2+x^{2n}+z_{ik^2}=0
\end{equation}
```

$$y^2 + x^{2n} + z_{ik^2} = 0 \quad (1)$$

összefüggés.

A fenti képletre magyar szövegben, feltéve, hogy magyar a kiválasztott nyelv, mondat elején az `\Aref({eq10})` parancs segítségével lehet hivatkozni, mondat közben pedig az `\aref({eq10})` parancsot használjuk.

Mindkét parancs kiteszi a határozott névelőt is, mégpedig a képletszámhoz igazított módon:

```
\Aref({eq10}) egyenlet példa a kitevők szedésére \ldots =
```

= Az (1) egyenlet példa a kitevők szedésére ...

vagy

A kitevők szedésének módját `\aref({eq10}) egyenlet szemlélteti\ldots =`
`= A kitevők szedésének módját az (1) egyenlet szemlélteti...`

A matematikai mód három fontos sajátossága:

1. Matematikai módban a \LaTeX elnyomja a szóköz karaktereket, ezek csak arra valók, hogy szedéskor jobban át tudjuk tekinteni a képlet szerkezetét. A helykihagyásokat a \LaTeX maga állapítja meg a képlet alapján. Ha szükséges akkor külön parancsokkal biztosíthatunk további helykihagyást.
2. Minden képlet pontosan egy bekezdés, és egy képlet sem tartalmazhat üres sort.
3. A képlet betűit matematikai változóként szedi a \LaTeX . Képletbeli közönséges szöveg csak speciális parancsokkal, pl. az `\mbox{szöveg}` parancs segítségével szedhető.

3.2. A képletek elemei

3.2.1. Görög betűk

Görög kisbetűk

α <code>\alpha</code>	θ <code>\theta</code>	o <code>o</code>	τ <code>\tau</code>
β <code>\beta</code>	ϑ <code>\vartheta</code>	π <code>\pi</code>	υ <code>\upsilon</code>
γ <code>\gamma</code>	ι <code>\iota</code>	ϖ <code>\varpi</code>	ϕ <code>\phi</code>
δ <code>\delta</code>	κ <code>\kappa</code>	ρ <code>\rho</code>	φ <code>\varphi</code>
ϵ <code>\epsilon</code>	λ <code>\lambda</code>	ϱ <code>\varrho</code>	χ <code>\chi</code>
ε <code>\varepsilon</code>	μ <code>\mu</code>	σ <code>\sigma</code>	ψ <code>\psi</code>
ζ <code>\zeta</code>	ν <code>\nu</code>	ς <code>\varsigma</code>	ω <code>\omega</code>
η <code>\eta</code>	ξ <code>\xi</code>		

Görög nagybetűk

Γ <code>\Gamma</code>	Λ <code>\Lambda</code>	Σ <code>\Sigma</code>	Ψ <code>\Psi</code>
Δ <code>\Delta</code>	Ξ <code>\Xi</code>	Υ <code>\Upsilon</code>	Ω <code>\Omega</code>
Θ <code>\Theta</code>	Π <code>\Pi</code>	Φ <code>\Phi</code>	

3.2.2. Indexek

Indexek az `^{\text{felső indexek}}` vagy az `_{\text{alsó indexek}}` módon szedhetők. Egy betűből álló index esetén elhagyható a kapcsos zárójel.

```
\begin{equation*}
(x_{0}+y_{0})^{\{n+1\}}=
(x_{0}+y_{0})^{\{n\}}(x_{0}+y_{0})
\end{equation*}
```

$$(x_0 + y_0)^{(n+1)} = (x_0 + y_0)^n (x_0 + y_0)$$

A

`\sideset{előtte}{utánna} \in amsmath`

parancs segítségével nagy operátorok előtt és után lehet indexeket elhelyezni.

```
\begin{equation*}
\sideset{_{a}^{\{b\}}}{_{lenn}^{\{fenn\}}}\prod
\end{equation*}
```

$${}_a^b \prod_{lenn}^{fenn}$$

3.2.3. Gyökjelek

Gyökjelek az

`\sqrt[n]{argumentum}`

parancs segítségével szedhetők:

`\begin{equation*}`

`\sqrt{x^2+2xy+y^2}=x+y`

`\end{equation*}`

$$\sqrt{x^2 + 2xy + y^2} = x + y$$

vagy

`\begin{equation*}`

`\sqrt[3]{x^3+3x^2y+3y^2x+x^3}=x+y`

`\end{equation*}`

$$\sqrt[3]{x^3 + 3x^2y + 3y^2x + x^3} = x + y$$

A gyökjel maga a `\surd` paranccsal szedhető ki: `\surd a = \sqrt{a}`

3.2.4. Aláhúzás, föléhúzás, kapcsos zárójelek

Aláhúzás és föléhúzás. A vonatkozó parancsok

`\overline{részképlet}` és `\underline{részképlet}`

alakúak. Pl.:

`\begin{equation*}`

`\overline{\overline{x}^2+\underline{ab}}%`

`+\overline{\overline{y}}}`

`\end{equation*}`

$$\overline{\overline{x^2 + ab} + \overline{y}}$$

Kapcsos zárójelek. A vonatkozó parancsok

`\overbrace{részképlet}^{szöveg a képletrész felett}` és

`\underbrace{részképlet}_{szöveg a képletrész alatt}`

alakúak. Pl.:

`\begin{equation*}`

`\underbrace{a+\overbrace{b+\cdots+f}^{\text{néhány kisbetű}}}`

`{\mbox{\small néhány kisbetű}}%`

`+g}_{\alpha \beta \gamma}`

`\end{equation*}`

$$a + \underbrace{b + \cdots + f}_{\alpha\beta\gamma} + g$$

3.2.5. Matematikai ékezetek

Matematikai ékezetek

`\hat{a}` `\breve{a}` `\grave{a}` `\bar{a}` `\check{a}`

`\acute{a}` `\tilde{a}` `\vec{a}` `\dot{a}` `\ddot{a}`

Az *i* és *j* betűk fölé úgy kell tenni vektorjelet, hogy ne legyen pont a betű felett:

`\vec{\imath}+\vec{\jmath}=0` `\vec{i}+\vec{j}=0`

Ha a vektor neve nemcsak egy karakterből áll akkor az

`\overrightarrow{jobbra nyíl e szöveg felett}` és a

`\overleftarrow{balra nyíl e szöveg felett}`

parancsokat érdemes használni. Pl.:

`\overrightarrow{AB}+\overleftarrow{AB}=0` `\overrightarrow{AB} + \overleftarrow{AB} = 0`

Ha be van töltve az `amsmath` csomag, akkor az alábbi matematikai ékezeteket tanácsoljuk használni:

Matematikai ékezetek $\in \backslash\text{amsmath}$

\hat{a} <code>\Hat{a}</code>	\breve{a} <code>\Breve{a}</code>	\grave{a} <code>\Grave{a}</code>	\bar{a} <code>\Bar{a}</code>	\check{a} <code>\Check{a}</code>
\acute{a} <code>\Acute{a}</code>	\tilde{a} <code>\Tilde{a}</code>	\vec{a} <code>\Vec{a}</code>	\dot{a} <code>\Dot{a}</code>	\ddot{a} <code>\Ddot{a}</code>
	$\overset{\cdot}{a}$ <code>\dddot{a}</code>	$\overset{\cdot\cdot}{a}$ <code>\dddot{a}</code>		

Az összehasonlítás kedvéért tekintsük az alábbi példákat:

$\hat{\hat{A}}$ <code>\hat{\hat{A}}</code>	$\hat{\hat{A}}$ <code>\Hat{\Hat{A}}</code>
$\vec{\ddot{a}}$ <code>\ddot{\vec{a}}</code>	$\vec{\ddot{a}}$ <code>\Ddot{\Vec{a}}</code>

3.2.6. Két szélesebb matematikai ékezet

Szélesebb ékezet szedésére az

`\widetilde{képletrész}` és a `\widehat{képletrész}`

parancsok révén nyílik mód:

<code>\widetilde{xy}=\widehat{wz}</code>	$\widetilde{xy} = \widehat{wz}$
--	---------------------------------

3.2.7. Függvénynevek

A képletek betűit általában matematikai kurzív betűvel szedik. Ez alól kivételek a matematikai függvények nevei, melyeket antikva betűvel szednek. A vonatkozó parancsokat táblázatosan foglaltuk össze:

A matematikai függvények nevei:

<code>arccos</code>	<code>\arccos</code>	<code>arcsin</code>	<code>\arcsin</code>	<code>arctan</code>	<code>\arctan</code>	<code>arg</code>	<code>\arg</code>
<code>cos</code>	<code>\cos</code>	<code>cosh</code>	<code>\cosh</code>	<code>cot</code>	<code>\cot</code>	<code>coth</code>	<code>\coth</code>
<code>csc</code>	<code>\csc</code>	<code>det</code>	<code>\det</code>	<code>dim</code>	<code>\dim</code>	<code>exp</code>	<code>\exp</code>
<code>gcd</code>	<code>\gcd</code>	<code>hom</code>	<code>\hom</code>	<code>inf</code>	<code>\inf</code>	<code>lim</code>	<code>\lim</code>
<code>lim inf</code>	<code>\liminf</code>	<code>lim sup</code>	<code>\limsup</code>	<code>ker</code>	<code>\ker</code>	<code>lg</code>	<code>\lg</code>
<code>ln</code>	<code>\ln</code>	<code>log</code>	<code>\log</code>	<code>max</code>	<code>\max</code>	<code>min</code>	<code>\min</code>
<code>sec</code>	<code>\sec</code>	<code>sin</code>	<code>\sin</code>	<code>sinh</code>	<code>\sinh</code>	<code>sup</code>	<code>\sup</code>
		<code>tan</code>	<code>\tan</code>	<code>tanh</code>	<code>\tanh</code>		

A felsorolt függvények határértékének szedése a

`\lim_{x\to\infty}`

parancs értelemszerű alkalmazásával történhet. Az eredményt szövegközi matematikai módban:

$\lim_{x\rightarrow 0} \sin(x)$

és kiemelt matematikai módban:

<code>\[\lim_{x\to 0} \sin(x) \]</code>	$\lim_{x\rightarrow 0} \sin(x)$
--	---------------------------------

különbözik egymástól.

3.2.8. Törtek és binomiális együtthatók

Matematikai egyenletekben a

<code>\frac{számláló}{nevező}</code>	
<code>\dfrac{számláló}{nevező}</code>	$\in \text{amsmath}$
<code>\tfrac{számláló}{nevező}</code>	$\in \text{amsmath}$

parancsok segítségével szedhetünk törteket. Pl.:

```

\begin{equation*}
\frac{\frac{a}{x-y}+\frac{b}{x-y}}{1+\frac{a-b}{a+b}}\quad \frac{\frac{a}{x-y}+\frac{b}{x-y}}{1+\frac{a-b}{a+b}}
\frac{\frac{a}{x-y}+\frac{b}{x-y}}{1+\frac{a-b}{a+b}}\quad \frac{\frac{a}{x-y}+\frac{b}{x-y}}{1+\frac{a-b}{a+b}}
\end{equation*}

```

Binomiális együtthatók a

`\fenn \choose lenn` és a `\fenn \atop lenn`

parancsok segítségével szedhetők ki. Az első esetben van zárójel, a második esetben nincs.

Pl.:

```

\[ \binom{r}{m} \binom{m}{k} = \binom{r}{k} \binom{r-k}{m-k}
\]

```

A fenti parancsok $\text{T}_{\text{E}}\text{X}$ parancsok. Ha be van töltve az `amstex` csomag használhatjuk a

`\binom{fenn}{lenn}` és a `\tbinom{fenn}{lenn}`, `\dbinom{fenn}{lenn} \in amsmath`

parancsokat:

```

\[ \binom{k}{1} \quad \tbinom{k}{1} \quad \dbinom{k}{1}
\]

```

Ezek előnye, hogy formailag is $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ parancsok.

3.2.9. Összegek, integrálok és más szimbólumok

Az alábbi szimbólumok mérete függ attól, hogy kiemelt vagy sorközi matematikai módban vagyunk:

Σ \sum <code>\sum</code>	\cap \bigcap <code>\bigcap</code>	\odot \bigodot <code>\bigodot</code>	\cup \bigcup <code>\bigcup</code>
\int \int <code>\int</code>	\oint \oint <code>\oint</code>	\otimes \bigotimes <code>\bigotimes</code>	\oplus \bigoplus <code>\bigoplus</code>
\prod \prod <code>\prod</code>	\coprod \coprod <code>\coprod</code>	\sqcup \bigsqcup <code>\bigsqcup</code>	\uplus \biguplus <code>\biguplus</code>
\vee \bigvee <code>\bigvee</code>	\wedge \bigwedge <code>\bigwedge</code>		

Az integrálási határok illetve a produktumjel indexeinek elhelyezése ugyancsak függ attól, hogy sorközi vagy kiemelt matematikai módban vagyunk. Sorközi matematikai módban ezek mindig a szimbólum mellé kerülnek:

$$\int_a^b f(x) dx \quad \text{avagy} \quad \prod_{i=1}^n a_i$$

3.2.11. Kaligrafikus betűk

Az úgynevezett kaligrafikus betűk a

$$\mathcal{A}, \mathcal{B}, \mathcal{C}, \text{ stb.}, \mathcal{Z}$$

parancs segítségével szedhetők:

$$A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, P, Q, R, S, T, U, V, W, Z$$

3.2.12. Helykihagyások matematikai módban

A helyközöket a \LaTeX maga állapítja meg, ha matematikai módban vagyunk. A \LaTeX által választott helyközök a

$$\backslash, \backslash:, \backslash;, \backslash_ \quad \backslashquad \quad \backslashqqquad$$

parancsok segítségével változtathatók meg. A kihagyott helyköz balról jobbra haladva növekszik:

$$\begin{array}{ccccccc} \backslash a b & \backslash a \backslash, b & \backslash a \backslash: b & \backslash a \backslash; b & \backslash a \backslash_ b & \backslash a \backslashquad b & \backslash a \backslashqqquad b \\ a b & a b & a b & a b & a b & a b & a b \end{array}$$

Kis negatív helyköz a $\backslash!$ parancssal szedhető. Hasonlítsa össze ezt az egyenletet:

```
\begin{equation*}
```

```
I=\int\int_{\mathcal{A}} f(x,y)\backslash, \mathrm{d} x\backslash, \mathrm{d} y
```

```
\end{equation*}
```

$$I = \int \int_{\mathcal{A}} f(x, y) dx dy$$

ezzel az egyenlettel

```
\begin{equation*}
```

```
I=\int!\!\!\int_{\mathcal{A}} f(x,y)\backslash, \mathrm{d} x\backslash, \mathrm{d} y
```

```
\end{equation*}
```

$$I = \iint_{\mathcal{A}} f(x, y) dx dy$$

3.3. Többsoros képletrészek és képletek

3.3.1. Mátrixok

Mátrixok a

$$\begin{array} & \end{array}$$

környezet segítségével szedhetők. Ez a környezet megegyezik paraméterezését illetően a táblázatok szedése kapcsán már megismert `tabular` környezettel. Az `array` környezet csak matematikai módban használható. A példa egy lineáris egyenletrendszer szedésén keresztül mutatja a környezet használatát. Érdeemes megfigyelni és elemezni az oszlopelrendezéssel kapcsolatos `{*{3}{c@{\:}\:}} c@{\;=\;}` paraméterek szerepét. Ez többek között, biztosítja, hogy a $+$ műveleti jelek, valamint az egyenlőségjelek egy oszlopban legyenek.

```
\begin{equation*}
```

```
\begin{array}{*{3}{c@{\:}\:}} c@{\;=\;}
```

```
a_{11}x_1 & a_{12}x_2 & \cdots & a_{1n}x_n & b_1 \\
```

```
a_{21}x_1 & a_{22}x_2 & \cdots & a_{2n}x_n & b_2 \\
```

```
\multicolumn{5}{c}{\dotfill} \\
```

```
a_{n1}x_1 & a_{n2}x_2 & \cdots & a_{nn}x_n & b_n \\
```

```
\end{array}
```

```
\end{equation*}
```

$$a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n = b_1$$

$$a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n = b_2$$

$$\cdots$$

$$a_{n1}x_1 + a_{n2}x_2 + \cdots + a_{nn}x_n = b_n$$

További példaként szolgál az alábbi mátrix:


```

\begin{equation*}
\mathbf{A} = \left[ \begin{array}{ccc}
x_{11} & x_{12} & \dots \\
x_{21} & x_{22} & \dots \\
\vdots & \vdots & \ddots
\end{array} \right]
\end{equation*}

```

3.3.2. Többsoros egyenletek

Többsoros egyenletek a

```

\begin{eqnarray}
az\ első\ képletsor & \\\
a\ második\ képletsor\ stb. & \\\
az\ utolsó\ képletsor & \\
\end{eqnarray}
\begin{eqnarray*}
az\ első\ képletsor & \\\
a\ második\ képletsor\ stb. & \\\
az\ utolsó\ képletsor & \\
\end{eqnarray*}

```

környezetek segítségével szedhetők, ahol a * nélküli változatban minden sor automatikusan képletszámot kap, ha csak nincs elhelyezve a sorvégi \nonumber parancs. A sor szerkezetét pedig az alábbiak mutatják:

baloldali képletrész & középső képletrész & jobboldali képletrész

A baloldali képletrész jobbra, a középső képletrész középre, a jobboldali képletrész pedig balra igazított. A képletrészek által alkotott oszlopok között \arraycolsep a távolság. Megjegyezzük még, hogy a eqnarray környezetek önmagukban matematikai módba kapcsolnak. Példaként két többsoros képlet illusztrálja a mondottakat.

```

{\setlength{\arraycolsep}{2pt}}
\begin{eqnarray}
(x+y)(x-y) & = & x^2 - xy + xy - y^2 \nonumber \\
& & = x^2 - y^2 \quad (2) \\
(x+y)^2 & = & x^2 + 2xy + y^2 \quad (3)
\end{eqnarray}

\begin{eqnarray*}
y & = & a + b + c + d + \\
& & e + f + g \\
& & h + i + j
\end{eqnarray*}

```

Vegyük észre, hogy második esetben nagyobb az oszlopok között távolság. Figyeljük meg az üres kapcsolószerűjel pár szerepét is.

3.3.3. Betűméret matematikai módban

Négy alapméret lehetséges:

1. `\displaystyle` – ez a kiemelt matematikai mód betűmérete;
2. `\textstyle` – ez a szövegközi matematikai mód betűmérete;
3. `\scriptstyle` – ez az indexbetű nagyságának megfelelő betűméret;
4. `\scriptscriptstyle` – ez az indexbetű indexének megfelelő betűméret.

Adott környezetben kapcsolószerűjel párba helyezve érvényesíthető a parancs. A parancs használatát két példa szemlélteti:

```
\begin{equation*}
\frac{\frac{a}{b}}{\frac{c}{d} + \frac{f}{g}}
\end{equation*}
```

vagy pedig

```
\begin{equation*}
\frac{\frac{a}{b}}{\frac{c}{d} + \frac{f}{g}}
\end{equation*}
```

3.4. Keretezett egyenletek kiemelt matematikai módban

Két megoldást mutatunk be. Az első esetben az `fbox` parancsot használjuk. Mivel ez vissza-kapcsol szövegmódba újra gondoskodni kell arról, hogy matematikai módban írassuk a képletet (`\fboxsep` a keret szövegtől mért távolsága):

```
\setlength{\fboxsep}{3pt}
\begin{equation}
\fbox{$\displaystyle \int_0^\infty f(x)\, dx \approx \sum_{i=0}^n w_i f(x_i)$}
\end{equation}
```

$$\int_0^\infty f(x) dx \approx \sum_{i=0}^n w_i f(x_i) \quad (4)$$

Ha be van töltve az `amsmath` csomag, akkor a `boxed` parancs használható egyenletek bekeretezésére:

```
\begin{equation}
\boxed{\int_0^\infty f(x)\, dx \approx \sum_{i=0}^n w_i f(x_i)}
\end{equation}
```

$$\int_0^\infty f(x) dx \approx \sum_{i=0}^n w_i f(x_i) \quad (5)$$

3.5. Kövér betűk, szimbólumok

Ha be van töltve az `amsmath` csomag, akkor a matematikai környezetben kiadott

```
\mathbf{betű(sorozat)}
```

parancs álló vastag betűkkel szedi a betűsorozat betűit. Dúlt vastag betű a

```
\boldsymbol{betű(sorozat)}
```

parancs segítségével szedhető. Ha nincs betöltve az `amsmath` csomag, akkor a `\boldmath` parancs használatát javasoljuk. Mivel ez a parancs csak szövegmódban adható ki, az `\mbox` paranccsal szövegmódba térünk át, itt adjuk ki a parancsot, majd matematikai módba lépünk vissza. Nyilvánvaló, hogy csak az `\mbox` parancs által létrehozott környezetében marad érvényes a `\boldmath` parancs. Csökkenti az írásmunkát, egy alkalmas parancsdefiníció. Ezt a preambulumban helyezzük el:

```
\newcommand{\BM}[1]{\mbox{\boldmath$#1$}}
```

Az alábbiak jól illusztrálják a lehetőségeket:

```

\begin{equation}
\rho,\; \nabla,\; M,\; \mathcal{T},\quad
\mathbf{\rho},\; \mathbf{b},\; \mathbf{M},\; \mathcal{T}
\end{equation}

```

$$\rho, \nabla, M, \mathcal{T}, \quad \rho, \mathbf{b}, \mathbf{M}, \mathcal{T} \quad (6)$$

avagy

```

\begin{equation}
\rho,\; \nabla,\; M,\; \mathcal{T},\quad
\boldsymbol{\rho},\; \boldsymbol{b},\; \boldsymbol{M},\; \boldsymbol{\mathcal{T}}
\end{equation}

```

$$\rho, \nabla, M, \mathcal{T}, \quad \boldsymbol{\rho}, \boldsymbol{b}, \boldsymbol{M}, \boldsymbol{\mathcal{T}} \quad (7)$$

és

```

\begin{equation}
\rho,\; \nabla,\; M,\; \mathcal{T},\quad
\rho,\; \nabla,\; M,\; \mathcal{T},\quad
\mathbf{BM}\{\rho\},\; \mathbf{BM}\{b\},\; \mathbf{BM}\{M\},\; \mathbf{BM}\{\mathcal{T}\}
\end{equation}

```

$$\rho, \nabla, M, \mathcal{T}, \quad \boldsymbol{\rho}, \boldsymbol{b}, \boldsymbol{M}, \boldsymbol{\mathcal{T}} \quad (8)$$

3.6. Az amsmath által nyújtott matematikai környezetek

A multiline környezet kiemelt matematikai módba kapcsol és egy többsoros egyenlet szedését teszi lehetővé.

```

\begin{multiline}
  Első képletsor      \ \ % Ez a képletsor balra igazított.
  Második képletsor   \ \ % A többi képletsor, az utolsót kivéve
  Harmadik képletsor \ \ % középre igazított.
  Utolsó képletsor   \ \ % Az utolsó képletsor jobbra igazított.
\end{multiline}

```

A képletszám vagy az utolsó sor után (standard L^AT_EX dokumentumosztályok), vagy a legelső sor elején (AMS dokumentumosztályok) jelenik meg. A *-os verzió esetén nincs képletszám:

```

\begin{multiline*}
  Egymást követő képletsorok.
\end{multiline*}

```

Ha azt akarjuk, hogy a képletszámok ne változzanak, hanem a kis abc betűi segítsék a képletek megkülönböztetését, akkor a

```

\begin{subequations}
  Kiemelt egyenleteket tartalmazó szöveg.
\begin{subequations}

```

környezetet kell alkalmaznunk.

A gather környezet kiemelt matematikai módba kapcsol és egymás követő sorokban elhelyezett egyenletek szedését teszi lehetővé.

```

\begin{gather}
  Első egyenlet          \ \ %
  Második egyenlet       \ \ % Valamennyi egyenlet középre igazított
  Harmadik egyenlet      \ \ % a szedés során.
  Utolsó egyenlet \notag \ \ % Képletszám nélküli sor.
\end{gather*}

```

Valamennyi sor kap képletszámot, hacsak nincs kiadva a parancs. A *-os verzió esetén nincs képletszám:

```
\begin{gather*}
  Egymást követő sorokban szedett képletszám nélküli egyenletek.
\end{gather*}
```

Az alábbi példában mindhárom környezet megjelenik:

```
\begin{subequations}
\begin{multline}
a+b+c+d+e+f=\\
=i+j+k+l+m+n
\end{multline}
\begin{gather}
a_1=b_1+c_1\\
a_2=b_2+c_2-d_2+e_2\notag
\end{gather}
\end{subequations}
```

$$\begin{aligned}
 a + b + c + d + e + f &= \\
 &= i + j + k + l + m + n \quad (9a)
 \end{aligned}$$

$$\begin{aligned}
 a_1 &= b_1 + c_1 \\
 a_2 &= b_2 + c_2 - d_2 + e_2 \quad (9b)
 \end{aligned}$$

4. További lehetőségek

4.1. Betűtípusok

Szövegkiemelés: `\emph{Kiemelt szöveg}` = *Kiemelt szöveg*

Félkövér szöveg: `\textbf{Félkövér szöveg}` = **Félkövér szöveg**

Kurzív szöveg: `\textit{Kurzív szöveg}` = *Kurzív szöveg*

Kiskapitális Szöveg: `\textsc{Kiskapitális Szöveg}` = KISKAPITÁLIS SZÖVEG

Írógép betű: `\texttt{Írógép betű}` = Írógép betű

Dőlt betű: `\textsl{Dőlt betű}` = *Dőlt betű*

Groteszk betű: `\textsf{Groteszk betű}` = Groteszk betű

Antikva betű: `\textrm{Antikva betű}` = Antikva betű

4.2. Betűnagyság

<code>\tiny</code> (5pt)	<code>{\tiny A legeslegkisebb}</code>	A legeslegkisebb
<code>\scriptsize</code> (7pt)	<code>{\scriptsize Nagyon kicsi}</code>	Nagyon kicsi
<code>\footnotesize</code> (8pt)	<code>{\footnotesize Eléggé kicsi}</code>	Eléggé kicsi
<code>\small</code> (9pt)	<code>{\small Kicsi}</code>	Kicsi
<code>\normalsize</code> (10pt)	<code>{\normalsize Alapméret}</code>	Alapméret
<code>\large</code> (12pt)	<code>{\large Nagyobbacska}</code>	Nagyobbacska
<code>\Large</code> (14.4pt)	<code>{\Large Nagy}</code>	Nagy
<code>\LARGE</code> (17.28pt)	<code>{\LARGE Még nagyobb}</code>	Még nagyobb
<code>\huge</code> (20.74pt)	<code>{\huge Még ennél is nagyobb}</code>	Még ennél is nagyobb
<code>\Huge</code> (24.88pt)	<code>{\Huge Legnagyobb}</code>	Legnagyobb

5. Grafika

5.1. Ábrák elhelyezése

A \LaTeX -ben történő ábraelhelyezésről, a grafika megjelenítéséről külön könyvet írtak [2]. Magyarul a [3] könyv 249-255 oldalaira utalunk, angolul pedig a `\texmf\doc\graphics\epslatex.ps` fájl tanulmányozását ajánljuk. Az utóbbi része a MiKTeX 2.0 disztribúciónak és a

`\texmf\doc\latex\graphics`

könyvtárban található¹. Alábbiak csak a legegyszerűbb és elsősorban csak a MiKTeX -ben szükséges ismeretekre szorítkoznak.

Mielőtt áttekintenénk a grafikai parancsokat visszaidézzük a \LaTeX blokk és az eps ábra szerkezetét. Az előbbit

5.1. ábra. A \LaTeX blokk szerkezete

az 5.1 ábra szemlélteti. Az eps ábrák jellemzője az `un.` határoló doboz (bounding box). Ennek méreteit álló és elforgatott ábra esetén az 5.2 ábra szemlélteti. Leolvasható az ábráról, hogy álló ábraelhelyezés esetén – baloldali ábrarészlet – a **szélesség** [`width`] és a **magasság** [`height`] a határoló doboz két jellegzetes mérete. Ha azonban elforgatott ábráról van szó, akkor a **szélesség** [`width`] és a **teljes magasság** [`total height`] a határoló méretek. Az eps fájl valójában egy ascii

5.2. ábra. Az eps ábra mint \LaTeX blokk

fájl. A fájl elején megtalálhatók a határoló doboz baloldali alsó és jobboldali felső sarkának valamilyen, pl. a lap baloldali alsó sarkához kötött koordinátarendszerben vett és `un.` **bigpoint** [`1 bp=1/72 inch`] mértékegységben mért koordinátái. A vonatkozó sor

```
%%BoundingBox: 56 56 264 191
```

¹A 2.1. verziószámú disztribúció sajnos nem tartalmazza az említett ps fájlt.

alakú, ahol az első két szám a baloldali alsó sarok x és y a második két szám pedig a jobboldali két sarok x és y koordinátája.

Visszatérve ezek után a grafikai parancsok kérdésére kihangsúlyozzuk, hogy az ismertetésre kerülő parancsok mindegyike a `graphicx` csomag része, ennek betöltéséről tehát gondoskodni kell.

A `MiKTeX 2.0` és `2.1` verziók telepítés után mindössze háromfajta, azaz a
`ps`, `eps` és `wmf`
típusú grafikai fájlokat tudják kezelni. Képesek azonban
`pcx`, `png`, `tga` és `tif`
típusú grafikai fájlok megjelenítésére is, ha az alábbi programokat elhelyezzük a `texmf\miktex\bin` könyvtárban:

```
ppmtobmp.exe  {már telepítéskor bekerül az említett
                                   könyvtárba}
giftopnm.exe  {gif ábrák esetén szükséges}
pcxtoppm.exe  {pcx ábrák esetén szükséges}
pngtopnm.exe  {png ábrák esetén szükséges}
tगतoppm.exe  {tga ábrák esetén szükséges}
tifftopnm.exe {tif ábrák esetén szükséges}
```

A fenti fájlok közül az első kettőt e-mail-ben kiküldtük a tárgy hallgatóinak. A másik három fájl is letölthető a hálóról az `un. Comprehensive TeX Archive Network` röviden `CTAN` szerverekről (`Pl. ftp.dante.de`). A `systems\win32\miktex\util\netpbm` könyvtárban lévő `cab` fájlban kell keresni.

Ha tehát `ps`, `eps`, `wmf` vagy `gif` ábrákat kívánunk elhelyezni a szerkesztett dokumentumban, akkor feltételezzük, hogy a preambulum tartalmazza a következő parancsokat:

```
\usepackage{graphicx} % az ábrák kezelésére szolgáló csomag -- amint erre már fentebb
                       % utaltunk ezt a csomagot feltétlenül be kell tölteni.
\DeclareGraphicsRule{.wmf}{bmp}{}{} % az utasítás lehetővé teszi wmf ábrák kezelését
\DeclareGraphicsRule{.gif}{bmp}{}{} % az utasítás lehetővé teszi gif ábrák kezelését
```

(A jelen `LATEX` fájl preambuluma tartalmazza a fenti parancsokat)

A preambulumban kiadott

```
\DeclareGraphicsRule{Kiterjesztés}{Fajltípus}{Read-fájl}{Parancs}
```

parancs négy paramétere rendre:

1. Kiterjesztés (most `wmf` és `gif`.)
2. Fajltípus (most `bmp` – a kiterjesztés megszabta grafikai fájl belső konverzióval `bmp` típusú lesz a megjelenítéskor.)
3. Read-fájl – az ábrák méretét tartalmazó fájl, ha egyáltalán szükséges (a legtöbb esetben szükségtelen paraméter).
4. Parancs – rendszerparancs, amelynek révén pl. tömörített ábrafájlok is olvashatók (a legtöbb esetben ez is szükségtelen paraméter).

Az ábra dokumentumban történő elhelyezése az

```
\includegraphics*{kulcsok}{az ábrát tartalmazó fájl neve} illetve az
```

```
\includegraphics{kulcsok}{az ábrát tartalmazó fájl neve}
```

parancsok segítségével történhet. Megjegyezzük, hogy a két parancs között a

```
\includegraphics*{kulcsok}{ábrafájl}=\includegraphics{kulcsok, clip=true}{ábrafájl}
```

összefüggés áll fenn. A kulcsok csak `eps` ábrák esetén működnek minden esetben és kifogástalanul. A többi esetben ellenőrizzük, hogy valóban működik-e a választott kulcs. A kulcsok alakja:

```
kulcsnév=kulcsparaméterek vagy kulcsnév
```

A kulcsokat, a legutolsót kivéve, vessző választja el egymástól. A legfontosabb kulcsokat az alábbiak ismertetik:

width A megjeleníteni kívánt ábraszélesség a \LaTeX által elfogadott valamilyen mértékegységben. A hiányzó mértékegységet mindig "bigpoint"-nak veszi a rendszer ha hosszegységről van szó. A megjelenített grafika, ha szükséges, skálázással éri el ezt a szélességet. Ez a kulcs minden ábraformátum esetén működik.

height A megjeleníteni kívánt ábramagasság a \LaTeX által elfogadott valamilyen mértékegységben. A grafika, ha szükséges, skálázással éri el ezt a magasságot. Ez a kulcs minden ábraformátum esetén működik.

totalheight A teljes ábramagasság a \LaTeX által elfogadott valamilyen mértékegységben. Ez elforgatott ábra esetén, amint azt az 5.2 ábra szemlélteti, nem azonos az ábramagassággal.

scale Skálázási tényező, a nagyítás illetve kicsinyítés mértéke, nincs mértékegysége.

angle Az elforgatás szöge. Fokban kell megadni. A pozitív érték óramutató járásával ellentétesen, a negatív azzal ellentétesen forgat. Ha nincs **origin** megadva a referencia pont körül, ez az ábra bal alsó sarka, történik a forgatás. A

```
\includegraphics[angle=45]{Fig1.eps}
```

parancs 45 fokkal forgatja el az ábrát.

origin A forgatás e pont körül történik. Ha nem adjuk meg külön, akkor a referencia pont körül történik a forgatás. Az argumentum helyközzel elválasztott két szám, pl.:

```
origin = 1cm 1.5cm,
```

ahol a forgatás középpontjának x és y koordinátáit adják meg a számok a referencia ponthoz viszonyítva.

bb Az argumentum négy koordináta. Ezeket helyköz választja el egymástól. A négy koordináta a **ps**, és **eps** ábrákkal kapcsolatos un. határoló doboz (bounding box) méreteit határozza meg. Az első két szám a bal alsó sarokhoz, a második két szám a jobb felső sarokhoz tartozik.

natwidth, natheight Ismét a `|bb|` opció alternatívája. A `|natheight=h,natwidth=w|` kulcsok egyenértékűek a `|bb = 0 0 h w|` kulccsal.

viewport A `|viewport|` kulcsnak négy argumentuma van, ugyanúgy mint a `|bb|` kulcsnak. Ez esetben az ábra bal alsó sarka a KR kezdőpontja és a négy argumentum egy téglalapot határoz meg ebben a KR-ben.

trim A `|trim|` kulcsnak is négy argumentuma van. Ezek rendre megváltoztatják a kép méreteit, anélkül, hogy az ábra fizikai méretei megváltoznának. A pozitív értékek méretcsökkenést, a negatív értékek méretnövekedést eredményeznek. A sorrend baloldal, alsó oldal, jobboldal felső oldal.

clip A kulcs értéke 'true' vagy 'false' lehet. Első esetben csak a viewport-on belüli része jelenik meg az ábrának. Ha nincs viewport megadva akkor a bounding box a viewport.

Az ábrák lehetnek un. úszó illetve nem úszó ábrák. Úsztatott ábraelhelyezést a

```
\begin{figure}[hol] ábra \caption[rövid cím]{teljes cím}\label{címke}\end{figure}
```

környezettel hozhatunk létre. A környezet paraméterezése megegyezik a table környezet paraméterezésével, a részleteket illetően a 2.17 alszakaszra utalunk.

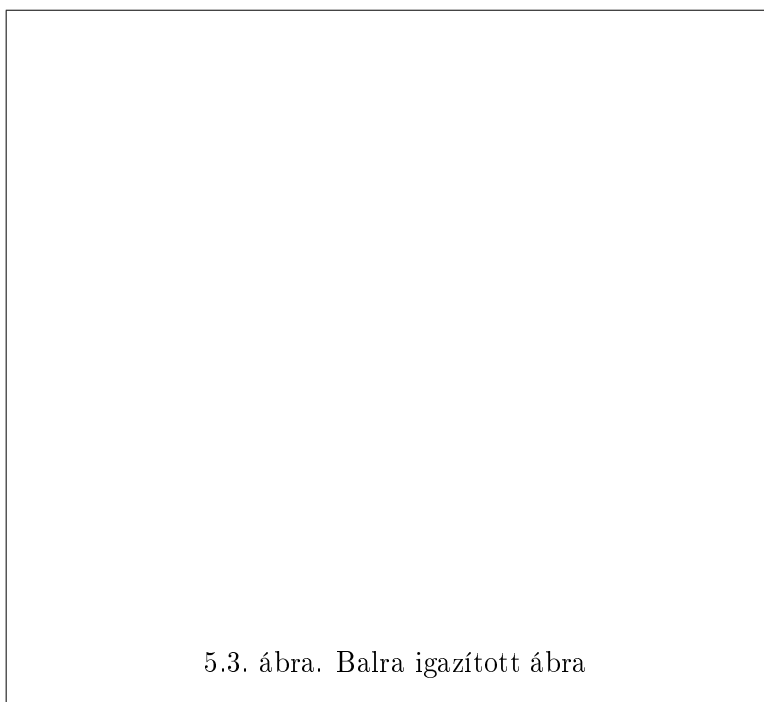
A rövid cím az ábrák jegyzékében jelenik meg, a teljes cím pedig az ábra alatt megjelenő és az ábra tartalmára utaló szöveg. A címke a hivatkozásokhoz szükséges.

5.1.1. WMF ábrák

Az alábbiak ábraelhelyezésekkel kapcsolatos néhány példát mutatnak be.

Balra igazított ábrát a

```
\parbox[b]{100mm}
{\begin{flushleft}
\includegraphics[
height=2.9378in,
width=3.1618in
% Ezek az eredeti méretek]
{3-4abra.wmf}%
\figcaption{Balra igazított
ábra}
\label{BIabra}
\end{flushleft}}%
```



parancsokkal hozhatunk létre – az ábra kerethez viszonyított elhelyezkedését figyeljük. A következő példa ugyanez az ábra középre igazítva:

```
\begin{center}
\includegraphics[
bb= 0in 0in
2.9378in 3.1618in]%
{3-4abra.wmf}
\figcaption{Középre
igazított ábra}
\label{CIabra}
\end{center}
```



Mindkét esetben eredeti méretében jelenik meg az ábra.

A címkék segítségével hivatkozhatunk: az 5.3 balra van igazítva, az 5.4 pedig jobbra.

Mivel az automatikus ábraszámok (táblázatszámok) létrehozása akkor is fontos lehet ha nem

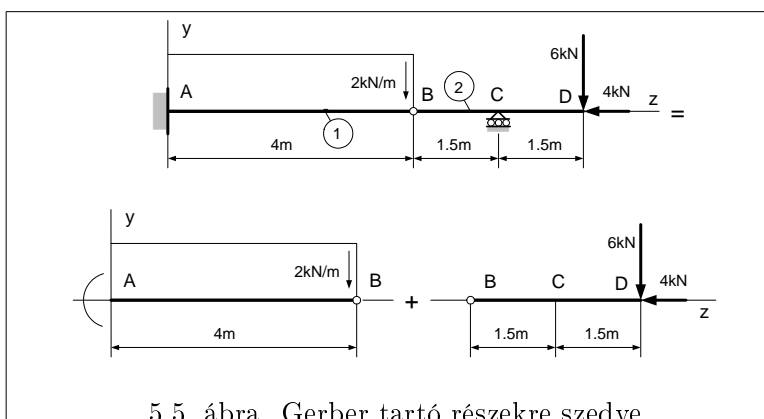
a `figure` (`table`) környezetet használjuk az ábra (vagy táblázat) elhelyezésekor, érdemes a

```
\makeatletter
\newcommand{\figcaption}{\def\@capytype{figure}\caption}
\newcommand{\tabcaption}{\def\@capytype{table}\caption}
\makeatother
```

parancsokat elhelyezni a preambulumban. Ezek használatával, lásd a fenti példákat, automatikusan sorszámozott ábrafelirat helyezhető el, és címke megadása esetén hivatkozni is lehet. (Táblázatok esetén a `tabcaption` parancsot használjuk.)

A következő ábra középre helyezett és eredeti méretének felére kicsinyített eps ábra.

```
\begin{figure}
\centering
\includegraphics[
bb = 0 0 486 259,
scale=0.5,
]{ZH3Fig8.eps}%
\figcaption{Gerber tartó
részekre szedve}
\label{gerber}
\end{figure}
```



5.1.2. GIF ábrák

5.6. ábra. Ez egy gif ábra

5.7. ábra. Ez egy másik gif ábra

6. Irodalomjegyzék

6.1. A thebibliography környezet

Irodalomjegyzéket a szöveg végén elhelyezett

```
\begin{thebibliography}{Címkeminta}
  bibliográfiai listelemek
\end{thebibliography}
```

listázó környezettel vagy bibliográfiai adatbázisok segítségével hozhatunk létre. Az egyes bibliográfiai listaelemeket a

```
\bibitem[címke]{kulcs} listaelem leírása
```

paranccsal szedhetjük ki.

Az opcionális [címke] hiánya esetén a \bibitem parancs minden egyes listaelemhez szögleteszárójelek közé helyezett számot szed. Ez a számsor eggyel kezdődik és egyesével növekszik. Ha azonban címke van jelen, akkor a szám helyett a címkével kezdődik a listaelem.

A kulcs a listaelem neve, a listaelemre (publikációra) pedig a

```
\cite{kulcs}
```

paranccsal hivatkozhatunk.

A jelen kézirat végén található irodalomjegyzéket a

```
\begin{thebibliography}{9}
\bibitem{Kopka} Helmut Kopka és Patrick W. Daly: \textit{A Guide to
\LaTeX\hspace*{-1.8mm}2$_{\varepsilon}$}, Második kiadás,
Addison--Wesley, 1995.
\bibitem{Goosens} Michael Goosens, Sebastian Rahtz és Frank Mittelbach:
\textit{The \LaTeX\hspace*{-1.8mm}2$_{\varepsilon}$ Graphics Companion,
Illustrating Documents with \TeX{} and Postscript}, Addison--Wesley, 1997.
\bibitem{Wettl} Wettl Ferenc, Mayer Gyula és Sudár Csaba:
\textit{\LaTeX\hspace*{-1.8mm}2$_{\varepsilon}$ kezdőknek és
haladóknak}, Panem kiadó, 1998.
\end{thebibliography}
```

módon szedtük ki.

6.2. Bibliográfiai adatbázisok és használatuk

6.2.1. A BIBTEX program

A BIBTEX egy segédprogram melynek segítségével automatikusan hozható létre az irodalomjegyzék egy vagy több ún. bibliográfiai adatbázisból. Az irodalomjegyzék létrehozása érdekében végre kell hajtani a

```
\bibliography{adatbázis1,adatbázis2,...}
```

L^AT_EX utasítást azon a ponton a dokumentumban, ahol meg kell jeleníteni az irodalomjegyzéket. A parancsban az adatbázis1 és adatbázis2 bib kiterjesztésű és az irodalomjegyzékkel kapcsolatos adatokat tartalmazó fájlok. Hivatkozás valamilyen publikációra most is a szövegben a hivatkozás kívánt helyén kiadott

```
\cite{kulcs}
```

paranccsal történik, ahol a kulcs a hivatkozott publikációt azonosító adat az adatbázisban (adatbázisokban). Maga az irodalomjegyzék elvben az alábbi három lépésben hozható létre:

1. Futtatjuk egyszer a L^AT_EX formatert a L^AT_EX dokumentumon.

2. A \LaTeX formater első futtatása után futtatjuk a \BIBTeX programot.
3. Ismét futtatjuk kétszer a hivatkozások kielégítése érdekében a \LaTeX formatert.

(A WinEdt32 szövegszerkesztőnek külön ikonja van a \LaTeX formater és a \BIBTeX segédprogram futtatására; a WinEdt 5.3 verziójában pedig a `texify` ikonra érdemes kattintani ez ugyanis automatikusan végrehajtja mindhárom lépést.)

Ha a \LaTeX dokumentumnak mondjuk `diplomamunka.tex` a neve, akkor a futtatások után kapunk egy `dokumentum.bbl` fájlt abban a könyvtárban ahol a dokumentumunk van. A `bbl` fájl valójában az irodalomjegyzéket tartalmazza. A `\bibliography{adatbázis1,adatbázis2,...}` parancs a `bbl` fájlt olvassa be irodalomjegyzékként a parancskiadás helyén.

Előfordulhat, hogy olyan publikációt is szeretnénk felvenni az irodalomjegyzékbe, amelyre nem hivatkozunk a szövegben. Ez a

```
\nocite{kulcs}
```

parancs kiadásával történhet. A parancs bárhol elhelyezhető a dokumentum szövegében.

Az irodalomjegyzék stílusát a

```
\bibliographystyle{stílus}
```

parancs határozza meg. Ez a parancs bárhol kiadható a dokumentumban, de meg kell előznie a `\bibliography{adatbázis1,adatbázis2,...}` parancsot. Mivel az irodalomjegyzék a legtöbb esetben a dokumentum végén van érdemes a fenti két parancsot közvetlenül az `\end{document}` parancs elé helyezni:

```
\bibliographystyle{stílus}
\bibliography{adatbázis1,adatbázis2,...}
\end{document}
```

A `stílus` paraméter négy alaptípusát és azok jellegzetességeit az alábbiak ismertetik:

- plain** Az irodalomjegyzék az első szerzők családneveinek megfelelően abc sorrendben rendezve jelenik meg. Minden egyes publikáció szögletes zárójelbe helyezett sorszámot kap. Ez a szögleteszárójelbe helyezett szám a `\cite` parancs kiadása helyén is megjelenik.
- unsrt** Az irodalomjegyzék a `\cite` és `\nocite` parancsok kiadásának sorrendjében jelenik meg. A sorszámok ugyanezen sorrendet követik mind az irodalomjegyzékben, mind pedig a hivatkozások helyein.
- alpha** Ugyanaz mint a **plain** de nem szám, hanem a szerzők nevéből és az évszámból álló rövidítés jelenik meg mind a szövegben, mind pedig a sorszámok helyein.
- abbrv** Ugyanaz mint a **plain** de a publikációk elemeit (szerzők nevei, hónapok nevei, folyóiratok címei) rövidítéssel jelennek meg.

A fenti négy mellett számos más bibliográfiai stílus létezik és magunk is hozhatunk létre bibliográfiai stílusokat a `makebst` program segítségével. A bibliográfiai stílusokat `bst` kiterjesztésű fájlok tartalmazzák. Ezeket a `texmf\bibtex\bst` könyvtárban találjuk és ott is keresi őket a rendszer.

6.2.2. A bibliográfiai adatbázis szerkezete

Első pillantásra úgy tűnhet, hogy érdekesebb a `thebibliography` környezetet használni irodalomjegyzék létrehozására. Ha jól meggondoljuk azonban, hogy a bibliográfiai adatbázisok (`bib` fájlok) létrehozása ugyanannyi gépelési munkát igényel mint a `thebibliography` környezetet létrehozása és azzal szemben ezek az adatbázisok ismételtlen felhasználhatók, akkor ez amellet szól, hogy inkább bibliográfiai adatbázisokat (`bib` fájlokat) használjunk. További előnye a bibliográfiai adatbázisoknak, hogy a stílus paraméter megszabta módon egységesen jelennek meg benne az azonos típusú publikációk.

A bib fájl egy ascii fájl amelyben úgynevezett bibliográfiai elemek találhatóak. A bibliográfiai elemek szintaxisát az alábbiak ismertetik:

```
@elemtípus{kulcs,
  mezőnév = {mezőszöveg},
  mezőnév = {mezőszöveg},
  .....
  mezőnév = {mezőszöveg},
  mezőnév = {mezőszöveg} }

@elemtípus{kulcs,
  mezőnév = "mezőszöveg",
  mezőnév = "mezőszöveg",
  .....
  mezőnév = "mezőszöveg",
  mezőnév = "mezőszöveg" }
```

Az elemtípusok külső {...} zárójelpárja helyett kerek (...) zárójelpár is szedhető.

6.2.3. A bibliográfiai elemek lehetséges típusai

A bibliográfiai elemtípusok mezői részint kötelezőek részint opcionálisak. Az alábbiak abc sorrendben ismertetik az elemtípusokat és a kötelező valamint opcionális mezőneveket.

@article Folyóiratban megjelenő cikk.

kötelező mezőnevek author, title, journal, year
opcionális mezőnevek volume, number, pages, month, note

@book Könyv és legfontosabb adatai.

kötelező mezőnevek author vagy editor, title, publisher, year
opcionális mezőnevek volume vagy number, series, address, edition, month, note

@booklet Nyomtatott, kötött kiadvány kiadó nélkül pl. prospektus.

kötelező mezőnevek title
opcionális mezőnevek author, howpublished, address, month, year, note

@inbook Egy könyv meghatározott része fejezete, szakasza etc.

kötelező mezőnevek author vagy editor, title, chapter és/vagy pages, publisher, year
opcionális mezőnevek volume vagy number, series, type, address, edition, month, note

@incollection Olyan könyvrészlet amelynek saját címe (szerzője) van.

kötelező mezőnevek author, title, booktitle, year
opcionális mezőnevek editor, volume vagy number, series, type, chapter, pages, address, edition, month, note

@inproceedings Konferencia kiadványban megjelenő cikk.

kötelező mezőnevek author, title, booktitle, year
opcionális mezőnevek editor, volume vagy number, series, pages, address, month, organization, publisher, note

@manual Műszaki dokumentáció.

kötelező mezőnevek title
opcionális mezőnevek author, organization, address, edition, month, note

@masterthesis Diplomaterv.

kötelező mezőnevek author, title, school, year
opcionális mezőnevek type, address, month, note

@misc Nem kategorizálható kiadvány.

kötelező mezőnevek Nincs ilyen.
opcionális mezőnevek author, title, howpublished, month, year, note

@phdthesis Doktori disszertáció.

kötelező mezőnevek author, title, school, year
opcionális mezőnevek type, address, month, note

@proceedings Konferenciakiadvány.

kötelező mezőnevek title, year
opcionális mezőnevek editor, volume vagy number, series, address, month, organization, publisher, note

@techreport Egy intézet vagy egyetem által készített műszaki jelentés.

kötelező mezőnevek author, title, institution, year
opcionális mezőnevek type, number, address, month, note

@unpublished Publikálatlan kiadvány.

kötelező mezőnevek author, title, note
opcionális mezőnevek month, year

Minden egyes elem tartalmazhat egy opcionális key mezőt, amely az abc sorrendbe történő rendezést segíti, ha hiányzik a szerzőre vonatkozó információ.

6.2.4. A mezők leírása

Az alábbiak abc sorrendben ismertetik az egyes mezők leírását.

address	A kiadó, vagy más intézet címe. Ismert kiadóknál elegendő a városnevet vagy a kötőjellel elválasztott városneveket megadni.
author	A szerző neve, vagy a szerzők nevei. (Keresztnév Vezetéknév vagy Vezetéknév, Keresztnév módon szedve. Több szerző esetén az angol and szó választja el a egymástól a szerzők neveit.
booktitle	Könyvcím azokban az esetekben, amikor csak a könyv egy részére (fejezetére, szakaszára etc.) történik hivatkozás.
chapter	Az idézett fejezet vagy szakasz száma.
edition	Információ arról hanyadik kiadásról van szó. Angolul a Second, Third etc. szavakat írjuk le, magyarul a Második, Harmadik etc. szavakat.
howpublished	A kiadás módjára utaló információ, ha nem a megszokott módon történik a kiadás, pl.: Magánkiadás.
institution	Annak az intézetnek a neve amelyik a műszaki jelentést szponzorálta.
journal	A folyóirat, vagy újság címe, ahol a vonatkozó cikk megjelent.
month	A megjelenés hónapja. Nem publikált anyagra történő hivatkozás esetén a megírás hónapja.
note	Bármiféle kiegészítő információ.
number	A folyóirat, újság, műszaki jelentés, vagy könyvsorozatban megjelenő könyv (sor)száma.
organization	A konferenciát etc. szervező intézmény, szervezet.
pages	Oldalszám. Pl. 37--41 vagy 55,58,72 vagy 65+ (az utóbbi forma azt jelenti, hogy 65 és magasabb oldalszámú oldalak).
publisher	A kiadó neve.
school	Az akadémiai intézmény neve, ahol az értekezés készült.
series	A könyvsorozat mint sorozat címe.

title	A munka címe. Közbülső szavak csak akkor kezdődnek nagybetűvel, ha kapcsolójelpárba helyezzük az első betűt.
type	A műszaki jelentés típusa. Pl. Zárójelentés.
volume	A folyóirat vagy könyv kötetszáma.
year	A munka megjelenésének, vagy publikálatlan esetben megírásának évszáma.

7. Vegyes dolgok

7.1. Új dokumentumosztály vagy stílus, közös néven csomag, hozzáadása

Elöljáróban megjegyezzük, hogy bár az ismertetésre kerülő eljárás minden esetben alkalmazható, a MiKTeX 2.1 esetén a MiKTeX Options varázsló külön lehetőséget biztosít ennek a feladatnak megoldására.

A követendő lépéseket az alábbiakban részletezzük:

1. A másolásra kiadott CD MiKTeX 210 könyvtárában megtalálható valamennyi csomag a neve alapján. Tegyük fel hogy a *listings* és a *fancyhdr* csomagokkal akarjuk bővíteni a rendszert. Tegyük fel továbbá, hogy a D meghajtón van a *localtexmf* könyvtár és legyen E a CD meghajtó neve. Helyezzük be a CD-t az E meghajtóba.
2. Hozzuk létre a


```
D:\localtexmf\tex\latex\listings és a
D:\localtexmf\tex\latex\fancyhdr
```

 könyvtárakat.
3. Keressük meg majd kattintsunk az


```
E:\miktex210\listings.cab
```

 fájlnevre, majd az `edit--select all` és `file--extract` kattintások után csomagoljuk ki a *listings.cab* fájl tartalmát a `D:\localtexmf\tex\latex\listings` könyvtárba.
4. Keressük meg majd kattintsunk az


```
E:\miktex210\fancyhdr.cab
```

 fájlnevre majd az `edit--select all` és `file--extract` kattintások után csomagoljuk ki a *fancyhdr.cab* fájl tartalmát a `D:\localtexmf\tex\latex\fancyhdr` könyvtárba.
5. Új csomag hozzáadásakor két eset lehetséges. (a) A *cab* fájl kicsomagolása után kapunk egy *ins* és *dtx* kiterjesztésű fájlt etc., de nincs *sty* vagy *cls* kiterjesztésű fájl a könyvtárban. Ekkor olvassuk be az *ins* kiterjesztésű fájlt a WinEdt vagy Texniccenter szövegszerkesztőbe és futtassuk rajta a L^AT_EX-et mintha *dvi* fájlt próbálnánk generálni. Eredményül a csomag hiányzó *sty* illetve *cls* fájljait kapjuk. (b) A kicsomagolás után megkapjuk a kívánt *sty* vagy *cls* kiterjesztésű fájlokat – most ez az eset forog fenn.
6. Indítsuk el a MiKTeX Options varázslót. A General, Refresh Now kattintásokkal hozzáférhetővé tehető a két új csomag.
7. A kicsomagolások után valamilyen formátumban egy-egy angolnyelvű leírás is található a létrehozott könyvtárakban – most *fancyhdr.pdf* vagy *listings.dvi* a vonatkozó fájlok nevei; ezek az Acrobat Reader illetve a YaP segítségével olvashatók. A csomagok felhasználása előtt mindenképpen érdemes áttanulmányozni a csomagok leírásait.

Más csomagok hozzáadása esetén is ugyanezeket a lépéseket kell követni.

A `fancyhdr` csomag segítségével a fejléc és lábléc megjelenését tudjuk testre szabni, széppé tenni.

A `listings` csomag jól használható programok listázására. Az alábbiak egy egyszerű *Pascal* lista esetén illusztrálják a `listings` csomag használatát. Maga a forráskód része lehet a \LaTeX fájlnek, vagy külön fájlként is beolvasható. Egy nagyon egyszerű példa kódját az alábbiak ismertetik:

```
\lstset{language=Pascal}           % Programozási nyelv kiválasztása
\begin{lstlisting}{}              % A listázó környezet meghívása
for i:= maxint to 0 do
begin
  {do nothing}
end;

Write('Case insensitive ');
Write('Pascal keywords.');
```

Az eredmény pedig:

```
for i:= maxint to 0 do
begin
  {do nothing}
end;

Write('Case_insensitive_');
```

Felhívjuk a figyelmet arra, hogy a `listings` csomag a legtöbb programozási nyelv esetén használható és a `lstlisting` környezetnek több a lista megjelenését befolyásoló, de itt nem ismertetett paramétere van. Ismételten hangsúlyozzuk, hogy érdemes kinyomtatni és áttanulmányozni csomag leírását adó `listings.dvi` fájlt.

Hivatkozások

- [1] Helmut Kopka és Patrick W. Daly: *A Guide to $\LaTeX_2\epsilon$* , Második kiadás, Addison–Wesley, 1995.
- [2] Michael Goosens, Sebastian Rahtz és Frank Mittelbach: *The \LaTeX_ϵ Graphics Companion, Illustrating Documents with \TeX and Postscript*, Addison–Wesley, 1997.
- [3] Wettl Ferenc, Mayer Gyula és Sudár Csaba: *\LaTeX_ϵ kezdőknek és haladóknak*, Panem kiadó, 1998.