

L^AT_EX matematika

Tételek, képletek, számozott formulák

Vadon Viktória

2023/24/I. félév

- 1 Tételek és definíciók
 - Tételek bevezető
 - Tételek alap \LaTeX -ben
 - `amsthm` csomag

Tételszerű környezetek

- céljuk: tételek, definíciók formázása tudományos cikkekben
- hogy néznek ki?
 - a beamer *különleges*, article-ben nem lesz színes, de beamer példa:

Theorem (Ez a tétel neve)

Ide jön a tétel szövege...

Akár több bekezdés, matematikai formulák, ...

- számozással, mini címsorral, néha térközzel és különböző betűstílussal emelkedik ki a szöveggörnyezetből
 - számozott objektum, ezért hivatkozható!
 - nem úszik, de tördelhető – csak egy szépen formázott szöveg
 - csomagokkal személyre szabható, pl. tétel teljes szövege automatikusan dőlt betűs, stb.
- újrahasznosítható más célra is, pl. feladatok, kódrészletek vizuális kiemelése

Tételek alap \LaTeX -ben I

- preambulumban definiálhatók tétel típusok, pl.

```
\newtheorem{tet}{Tétel}
```

- **tet**: környezetnév – kódban ezzel utalunk rá
 - **Tétel**: megjelenítendő név
- opcionális argumentumok a számozásra:
 - számozás szakaszolási egységen, pl. section-ön belül

```
\newtheorem{tet}{Tétel}[section]
```
 - közös számozás (már definiált) másik tételszerű környezettel:

```
\newtheorem{defin}[tet]{Definíció}
```
- !! figyeljünk a különbségre a pozícióban!
- !! egyszerre csak az egyik opcionális argumentumot használjuk!
együtt számozással a (nem) section-ön belüli számozás is öröklődik

Tételek alap L^AT_EX-ben II

- dokumentumtörzsben egy-egy típusból több, pl. több tétel, több definíció is elhelyezhető, szintaxis:

<code>\begin{defin}</code>	<code>\begin{tet}[Pitagorasz]</code>
A derékszögű háromszög...	Pitagorasz tétele...
<code>\end{defin}</code>	<code>\end{tet}</code>

1. Definíció

A derékszögű háromszög...

2. Tétel (Pitagorasz)

Pitagorasz tétele...

- opcionális argumentum, példánkban `[Pitagorasz]`: a tétel (vagy definíció, stb.) neve
- jó tudni: ha listával kezdődik a tétel, `\leavevmode` paranccsal tudjuk új sorban kezdeni

amsthm csomag I

- amsthm csomaggal: személyre szabható megjelenés és bizonyítás környezet
- tételstílusok `\theoremstyle{stílus}`
 - preambulumban, a `\newtheorem` parancsok előtt használjuk, a következő `\theoremstyle` parancsig él (a köztük `\newtheorem`-mel definiált típusokra)
 - előre definiált stílusok:
 - `plain`: alapértelmezett, alap L^AT_EX tételstílus: címsor vastag betűs, szöveg dőlt betűs
 - `definition`: címsor vastag betűs, szöveg sima
 - `remark`: címsor dőlt betűs, szöveg sima
- számozatlan típusok `\newtheorem*` paranccsal
 - szintaxis azonos a `\newtheorem`-mel, de nincsenek opcionális argumentumai

amsthm csomag II

- bizonyítás: proof környezet, például:

```
\begin{proof}
```

Tudjuk, hogy...

```
\end{proof}
```

Bizonyítás. Tudjuk,
hogy... □

```
\begin{proof}[Pitagorasz %
```

```
tétel bizonyítása]
```

Tudjuk, hogy...

```
\end{proof}
```

Pitagorasz tétel bizonyítása.

Tudjuk, hogy... □

- babel-lel lefordítja a generált bizonyítás szót

!! opcionális argumentum ezt teljesen felülírja

- a bizonyítás végén automatikusan kiteszi a kis négyzetet: □

- alapból fekete, csak a beamer *különleges*
- kézzel: `\qed` – *pluszban* kirakja akárhol
- *áthelyezés* proof-on belül: `\qedhere`

2 Képletek szintaxisa

- Ajánlott csomagok
- Képletek írásához – matematikai mód
- Formulák építése
 - Építőelemek
 - Építkezés
 - Szöveg dekoráció
 - Matematikai betűstílusok
 - Matematikai térközök

Ajánlott csomagok

- egyes matematikai parancsok (szimbólumok, formulák alap építőelemei, mint tört, szumma, stb.) alap \LaTeX -ben is élnek
- de ha bármi matematikához kapcsolódót írunk, érdemes kapásból betölteni:
- `amsmath`, `mathtools`
 - mindkettő hoz magával kifinomultabb formula-építőelemeket
 - `amsmath`: `display math` formulák számozása, igazítása
 - `mathtools`: saját makrókhoz lesz hasznos
- `amsfont`, `amssymb`
 - `amsfont` – betűkészletek, mint duplaszárú \mathbb{R} , kézírásos \mathcal{R} , gótikus \mathfrak{R} (később részletesen)
 - `amssymb` – extra szimbólumok (relációs jelek stb.)
- igénytől függően szükség lehet más csomagokra is

Képletek írásához – matematikai mód I

- képletek írása csak ún. *matematikai módban* lehetséges
 - gondoljunk rá úgy, mint egy speciális környezetre, ahol a képletet generáló parancsok élnek
 - \LaTeX fejében ettől kicsit bonyolultabb a dolog
- `verbatim`-hoz hasonlóan `inline` és „környezet” verziók
- *inline maths*, sorközi matematikai mód
 - cél: mondat közepén, szöveggel egy sorban változó, függvény, rövid képlet, stb.
 - ki-bekapcsolás: `$`
 - alternatív: kezdés `\(`, befejezés `\)`
 - például: `n négyzete n^2 = n négyzete n^2`
 - !! ne írjunk bele túl magas formulát, mert megnöveli a sormagasságot és/vagy olvashatatlanul kicsi lesz

Képletek írásához – matematikai mód II

- *display maths*, kiemelt matematikai mód \approx környezet
 - cél: hosszabb képletek
 - mindig külön sorba kerül, tipikusan középre zárt és fölötte-alatta extra térköz
 - kezdés: `\[`, befejezés `\]`
 - alternatív: ki-bekapcsolás: `$$` – ősz `TEX`, nem stabil
 - például `\[a^2 + b^2 = c^2 \]`

$$a^2 + b^2 = c^2$$

- a teljes képlet egyben kerüljön bele
- ha számolni is szeretnénk a *display* formulát, `\[`, `\]` helyett `\begin{equation}`, `\end{equation}`

Mire figyeljünk? I

matematikai mód sajátosságai, általános tanácsok

!! matematikai módban másképp dolgozza fel a szöveget:

- whitespace-t elnyeli, térközöket a \LaTeX határozza meg
 - csak szöveghez képest (is) másképp
 - pro: itt is úgy tagolhatjuk a kódot, ahogy nekünk átlátható
 - később látjuk majd, hogyan tudjuk mi beállítani a térközt
- minden betűt, ami nem parancs része vagy argumentuma, változónak tekint!
 - dőlt betű, furcsa térköz, ékezetes betűk nem használhatók!
 - **rendes szöveggént formázott szöveg** elhelyezése matematikai módban `\text{szöveg}` paranccsal

Mire figyeljünk? II

- következmény: ha mondatba ágyazott képleteket, változóneveket használunk, azok *kerüljenek* matematikai módba, de *csak* azok!
 - \LaTeX szemlélet – jelöljük a funkciót, és a \LaTeX e szerint formáz
 - jó példa:
Legyenek egy háromszög oldalai a , b és c
Legyenek egy háromszög oldalai a , b és c
 - rossz példa 1:
Legyenek egy háromszög oldalai a , b és c
Legyenek egy háromszög oldalai a , b és c
 - rossz példa 2:
Legyenek egy háromszög oldalai a , b és c
Legyenek egy háromszög oldalai a , b és c

Formulák építése – építőelemek I

- működnek a számok, billentyűzeten található műveleti és relációs jelek: =, +, -, <, >, /
- szorzás \cdot : `\cdot`, \times : `\times`, osztás \div : `\div`
- zárójelek: (,), [,] működnek; $\{ \}$: `\{ \}`, $\langle \rangle$: `\langle \rangle`; $|$ vagy $\|$: `\|`, $\lfloor \rfloor$: `\lfloor \rfloor`, $\lceil \rceil$: `\lceil \rceil`
- betűk = változóként formázva
- szó, szöveg beszúrása `\text{szöveg}` paranccsal
- egyes függvénynevek beépített paranccsal szépen formázva
 - pl. \sin : `\sin`, \lim : `\lim`
 - offline szerkesztőben menüből elérhetők
 - külön diasorban szimbólum referencia!

Formulák építése – építőelemek II

- szimbólumok beszurása
 - szimbólumok = görög betűk, relációs jelek, operátorok (mint szumma, integrál, stb.), stb.: \LaTeX parancsokkal
 - de nem kell fejből tudni őket!
 - offline szerkesztőkben szimbólum táblák, kattintással beszurja a kódját
 - Overleaf használóknak: Detexify (weblap link) – szimbólum rajzolható, felismeri és listázza a hasonlókat, \LaTeX parancsokkal (és szükséges csomaggal) együtt
 - vagy külön diasorban szimbólum referencia
 - néhány gyakori:
 - szumma $\backslash sum$: \sum , integrál $\backslash int$: \int
 - relációs jelek $\backslash leq$: \leq , $\backslash geq$: \geq
 - végtelen $\backslash infty$: ∞

Formulák építése – építkezés I

- index: felső index \wedge (kalap), alsó index $_$ (alulvonás) vezérlő karakterekkel:
 - $\$a^2\$$: a^2
 - $\$a_n\$$: a_n
 - ugyanaz a karakter/szimbólum kaphat alsó és felső indexet is (a két index sorrendje mindegy)
 $\$a_n^2\$$: a_n^2
 - ha több karakter az index, tegyük {}-be,
pl. $\$a_{12}\$$: a_{12}
 - függvény, szimbólum is kaphat indexet,
pl. $\$\log_2 a\$$: $\log_2 a$, $\$\alpha^2\$$: α^2
- derivált aposztróffal, $\$f'\$$: f'
- operátor határai, behelyettesítés is index-szel,
pl. $\$\int_0^\infty\$$: \int_0^∞ ,
 $\$f'(x)\vert_{x=0}\$$: $f'(x)|_{x=0}$

Formulák építése – építkezés II

- tört $\frac{a}{b}$: $\frac{a}{b}$
- gyök \sqrt{a} : \sqrt{a}
 - n-dik gyök $\sqrt[3]{a}$: $\sqrt[3]{a}$
- amsmath csomaggal binomiális együttható
- az argumentum önmagában lehet egy (komplex) formula, pl.

$$\left[\sum_{n=1}^{\infty} \frac{1}{n^2} \right]$$
$$= \frac{\pi^2}{6}$$

$$\sum_{n=1}^{\infty} \frac{1}{n^2} = \frac{\pi^2}{6}$$

- emeletes index csoporttal x^{n^2} : x^{n^2}

Szöveg dekoráció

Avagy hogy nem fogy ki soha a tudós a jelölésekből

Matematikai „ékezetek” – pl. vektor, stb. jelölésére

- egy változóra

- $\text{\textbackslash}\tilde{x}$ \tilde{x}
- $\text{\textbackslash}\bar{x}$ \bar{x}
- $\text{\textbackslash}\vec{x}$ \vec{x}
- $\text{\textbackslash}\hat{x}$ \hat{x}
- $\text{\textbackslash}\dot{x}$ \dot{x}
- stb.

- több változóra

- $\text{\textbackslash}\widetilde{abc}$ \widetilde{abc}
- $\text{\textbackslash}\widehat{abc}$ \widehat{abc}
- $\text{\textbackslash}\overline{abc}$ \overline{abc}
- $\text{\textbackslash}\underline{abc}$ \underline{abc}
- $\text{\textbackslash}\overrightarrow{abc}$ \overrightarrow{abc}
- $\text{\textbackslash}\overleftarrow{abc}$ \overleftarrow{abc}

Matematikai betűstílusok I

Avagy hogy nem fogy ki soha a tudós a jelölésekből

- alapértelmezés – változóként formázott betűk:
 $\text{\mathnormal{abcABC}}$ *abcABC*
 - *néhány* szövegben használatos betűstílusnak megvan a matematikai párja
 - ezek kb. $\text{\texttt{}}\text{\textit{}}\text{\textbf{}}\text{\textit{}}$ variánsai – nem változóként formázott szöveg, különböző betűstílusokkal, pl. függvénynevek formázására
 - $\text{\mathrm{abcABC}}$ *abcABC*
 - $\text{\mathsf{abcABC}}$ *abcABC*
 - $\text{\mathhtt{abcABC}}$ *abcABC*
 - $\text{\mathbf{abcABC}}$ **abcABC**
 - $\text{\mathit{abcABC}}$ *abcABC*
- !! *nem* halmozhatók – egymásba ágyazáskor a legbelső érvényesül

Matematikai betűstílusok II

Avagy hogy nem fogy ki soha a tudós a jelölésekből

- *extra* betűstílusok csak matematikai módbanÉ
 - többségük csak nagybetűkre!
- kalligrafikus betűk: \mathcal{ABC} *ABC*
- dupla szárú („blackboard bold”) \mathbb{NZZR} *NZZR* – `amsmath` csomag
 - indikátorhoz dupla szárú 1-es: `bbm` csomag, \mathbb{m}
- gótikus betűk \mathfrak{abcABC} *abcABC* – `amsmath` csomag
- kézírásos (extra kacskaringós) betűk \mathscr{ABC} *ABC* – `mathrsfs` csomag
- vastagon szedett görög betűk (és/vagy egyéb szimbólumok): `amsmath` csomaggal `\boldsymbol` parancs

Matematikai betűstílusok III

Vagyis hogy nem fogy ki soha a tudós a jelölésekből

- !! óvatosan a sok betűstílussal, mert csak korlátos számú betűkészletet/abécét enged betölteni
 - !! és szimbólumcsomagok is sokszor abécétet definiálnak
 - néha érdemes nem a teljes csomagot betölteni, csak egy-egy szimbólumparancs definícióját kimásolni
- betűtípusok lecserélése: `mathalpha` csomaggal
 - dokumentáció (link), benne betűtípus-minták(!)
 - használat pl.

```
\usepackage[cal=pptx,scr=boondoxo]{mathalpha}
```

`\mathcal` parancs a `pptx`, `\mathscr` parancs a `boondoxo` betűtípust használja
 - átállítható még `bb`, `frac`
 - egy-egy típus átméretezése, ha nem illik a dokumentumba, pl. `calscaled=0.95`
 - mind átméretezése `scaled=0.95`

Matematikai térközök

- térköz nélkül $\$ab\$$ ab
- kicsi $\$a\,b\$$ $a\ b$
- közepes $\$a\:b\$$ $a\ b$
- nagy $\$a\;b\$$ $a\ b$
- quad $\$a\quad b\$$ $a\quad b$
- qquad $\$a\qquad b\$$ $a\qquad b$
- negatív térköz (halmazható:) $\$a\!b\$$ ab

3 Méretezés

- Teljes formula méretezése
- Parancs variánsok különböző méretezéssel
- Zárójelek méretezése
- Szumma határai

Teljes formula méretezése I

- inline (sorközi) VS display (kiemelt) matematikai mód
nem csak a szöveghez képesti elhelyezésében különbözik, lásd:
- inline $\$ \sum_{k=1}^n \frac{1}{k} \$$: $\sum_{k=1}^n \frac{1}{k}$
- VS display $\left[\sum_{k=1}^n \frac{1}{k} \right]$:

$$\sum_{k=1}^n \frac{1}{k}$$

- itt látható különbségek:
 - szumma jel \sum mérete
 - szumma határainak elhelyezése
 - tört mérete

Teljes formula méretezése II

- megint változik a méretezés és elrendezés, ha egy formulát indexbe vagy törtbe ágyazunk! pl.

`\[\frac{\sum_{k=1}^n \frac{1}{k}}{n} \]`:

$$\frac{\sum_{k=1}^n \frac{1}{k}}{n}$$

- négyféle méretezési-elrendezési „stílus”
 - `\displaystyle` – display, kiemelt matematikai formula
 - `\textstyle` – inline, sorközi matematikai formula
 - `\scriptstyle` – index stílus
 - `\scriptscriptstyle` – index indexe (emeletes index) stílus
- display (kiemelt) formula nyilván `\displaystyle`-ről, inline (sorközi) formula `\textstyle`-ről indul

Teljes formula méretezése III

- komplex formulába beágyazott formula eggyel „alacsonyabb szintű” stílust kap
- a fenti parancsokkal másik stílus kényszeríthető a formulára/részeire! pl. tört számlálójában „nagy” szumma és tört:

```
\[ \frac{\displaystyle \sum_{k=1}^n
```

```
\frac{1}{k}}{n} \]
```

$$\frac{\sum_{k=1}^n \frac{1}{k}}{n}$$

- !! argumentum nélküli parancsok; hatáskör: formula végéig, felülírásig, vagy adott blokkon/parancs **argumentumon** belül

Parancs variánsok különböző méretezéssel I

- eredeti tört $\frac{\{ }\{ \}$
 - igazodik az aktuális méretezési-elrendezési stílushoz
- $\frac{\{ }\{ \}$ = $\displaystyle\frac{\{ }\{ \}$
- $\frac{\{ }\{ \}$ = $\textstyle\frac{\{ }\{ \}$
- lánc tört $\frac{\{ }\{ \}$
 - megtartja a kezdő méretezési-elrendezési stílust
 - több emeletes egymásba ágyazáshoz hasznos

Zárójelek méretezése I

- miért szükséges? például:

$$\backslash [2 (\frac{1}{a} + \frac{1}{b}) \backslash]$$
$$2\left(\frac{1}{a} + \frac{1}{b}\right)$$

- nem csak csúnya, nehéz átlátni és értelmezni is!
- automatikus méretezés: `\left(, \right)` *parancspár*
- például

$$\backslash [2 \backslash \text{left} (\frac{1}{a} + \frac{1}{b} \backslash \text{right}) \backslash]$$
$$2\left(\frac{1}{a} + \frac{1}{b}\right)$$

Zárójelek méretezése II

!! mindkettőhöz *kötelező* valamilyen zárójel

- ha csak az egyik oldali zárójel szükséges: a másik helyére egy pont .
- pl. függvényérték helyettesítése:

```
\[ \left.\frac{e^x}{x} \right|_{x=2} \]
```

$$\left. \frac{e^x}{x} \right|_{x=2}$$

!! *mindig* párban kell lenniük

- ugyanazon blokkon/argumentumon belül!
- (többsoros formulában, ld. később) egy soron belül

Zárójelek méretezése III

- alternatív: **kézi** méretezés
- *miért?*
- nem feltétlen kell párban állniuk!
 - magában álló zárójel, vagy külön blokkba, pl. színezés, külön sor miatt
- automatikus méretezés nem mindig jó méretet választ!
- pl. több egymásba ágyazott zárójelnél...
 - **automatikus** méretezés

```
\[ F \left( 2 \left( \frac{1}{a} + \frac{1}{b} \right) \right)
```

$$F \left(2 \left(\frac{1}{a} + \frac{1}{b} \right) \right)$$

- ezzel szemben **kézzel**:

Zárójelek méretezése IV

```
\[ F \biggl( 2 \Bigl( \frac{1}{a} + \frac{1}{b} \Bigr) \biggr) \]
```

$$F\left(2\left(\frac{1}{a} + \frac{1}{b}\right)\right)$$

- parancsai – növekvő méretek:

- `\big` < `\Big` < `\bigg` < `\Bigg`
- normáltól `\Bigg`-ig:

```
$)\bigr)\Bigr)\biggr)\Biggr)$
```

```
)))
```

- **bal**, **közép** és **jobb** verziók, pl. `\bigl`, `\bigm` és `\bigr`
 - külön-külön is használhatók, vagy csak `l+r`
 - térközben különböznek, igyekezzünk a jó verziót használni

Szumma és egyéb operátorok határai I

- szumma megjelenése változik inline (sorközi) és display (kiemelt) matematikai mód között:

- $\sum_{k=1}^n$
- $\displaystyle\sum_{k=1}^n$

$$\sum_{k=1}^n$$

- itt a `\displaystyle`, `\textstyle` parancsok egyszerűen változtatják a méretet és a határok elhelyezését!
- ez egyéb operátorokra is igaz!

- produktum $\prod_{k=1}^n$
- $\displaystyle\prod_{k=1}^n$

Szumma és egyéb operátorok határai II

- integrál \int_1^2 : \int_1^2 , \int_1^2

hasonlóképp \oint , \iint , stb.

!! az integrál kivétel, alapértelmezésben a nagy integráljel is mellette kapja a határait

- amsmath csomag opcióval állítható át:
`\usepackage[intlimits]{amsmath}`
- unió $\bigcup_{k=1}^n$,
 $\bigcup_{k=1}^n$, hasonlóképp
metszet \bigcap
- határérték: $\lim_{n \rightarrow \infty}$,
 $\lim_{n \rightarrow \infty}$, hasonlóképp
 \liminf : \liminf , \inf : \inf , \min : \min , stb.
- stb.

Szumma és egyéb operátorok határai III

- ha *csak* a határok elhelyezését szeretnénk módosítani:
- `\limits`, `\nolimits` parancsok, *közvetlen az operátor után, az indexek elé beszúrva*

- `\lim\limits_{n\to\infty}`
`\sum\limits_{k=1}^n \frac{1}{k^2}` `\$`

$$\lim_{n \rightarrow \infty} \sum_{k=1}^n \frac{1}{k^2}$$
- `\left[\lim\nolimits_{n\to\infty}`
`\sum\nolimits_{i=1}^n \frac{1}{k^2} \right]`

$$\lim_{n \rightarrow \infty} \sum_{k=1}^n \frac{1}{k^2}$$

Többsoros és hosszú index |

- többsoros index: `\substack` parancs (`amsmath` csomag):

```
\[ \sum_{\substack{n \in \mathbb{N} \\ n \text{ páros}}} \]
```

$$\sum_{\substack{n \in \mathbb{N} \\ n \text{ páros}}}$$

- sortörés benne `\\` paranccsal

Többsoros és hosszú index II

- ha hosszú a határ, aszerint hagy térközt, például:

$$\sum_{a \in \mathbb{R}} f(a) \quad \text{VS} \quad \sum_{a, b, c \in \mathbb{R}} f(a, b, c)$$

- megoldások:

- kézzel negatív térköz \! (akár halmozva is)
- vagy `\mathclap` paranccsal (mathtools csomag)
- úgy tesz, mintha 0 szélességű lenne az index:

```
\[ \sum_{\mathclap{a, b, c \in \mathbb{R}}} f(a, b, c) \]
```

$$\sum_{a, b, c \in \mathbb{R}} f(a, b, c)$$

- óvatosan vele, ha a szumma mellé is magas formula, pl. tört kerül, akkor rányomtathatja az indexet!

- ## 4 Komplex formulák
- Mátrix
 - Esetszétválasztás
 - Kapocs, magyarázat...
 - Relációs jel „dekoráció”

Mátrix I

- amsmath csomaggal matrix környezet

- példa:

- ```
\[A = \left(\begin{matrix} 1 & 2 & 0 \\ 16 & -1 & 5 \\ 1 & 1 & 4.5 \end{matrix} \right) \]
```

$$A = \begin{pmatrix} 1 & 2 & 0 \\ 16 & -1 & 5 \\ 1 & 1 & 4.5 \end{pmatrix}$$

- szintaxis: mint tabular
- cellák elválasztása & jellel, sor lezárása \\
- de nem kell előre megadni az oszlopok számát és típusát :)
- kerülhet egyéb formulákkal egy sorba, függőlegesen középre igazodik

# Mátrix II

- zárójelek:
  - `matrix` zárójel nélküli verzió, zárójelezés kézzel, `\left( \right)`-tal
  - zárójeles variánsok (szintén `amsmath` csomag):
  - `pmatrix`- parantheses ( )
  - `bmatrix`- brackets [ ]
  - `Bmatrix`- braces { }
  - `vmatrix`- vert |
  - `Vmatrix`- Vert ||
- oszlopok igazítása:
- `amsmath` `matrix`-ai középre igazítanak
- `mathtools` csomaggal `\begin{matrix*}[r]`: opcionális argumentumba az *összes oszlop egységes igazítása*
  - pl. `l`, `r`, `c`, vagy *bármilyen* definiált oszloptípus!
  - `pmatrix*`, stb. verziók is!

## Mátrix III

- például

```
\[\begin{pmatrix} r
1 & 2 & \\
16 & -1 & 5 \\
1 & 1 & 4.5 \\
\end{pmatrix} \]
```

$$\begin{pmatrix} 1 & 2 & 0 \\ 16 & -1 & 5 \\ 1 & 1 & 4.5 \end{pmatrix}$$

- még nagyobb szabadság: array környezet
  - tabular matematikai verziója
  - analóg szintaxis, oszlopok deklarálása előre, különböző igazítással, rácsvonalak, stb.



## Mátrix IV

- kicsinyített mátrix: `smallmatrix` környezet – `mathtools` csomag
  - `psmallmatrix`, `psmallmatrix*`, stb. verziók is
    - \*-os verziónál csak `l`, `r`, `c` igazítás
  - méretek összehasonlítóképp:

```
\[\begin{psmallmatrix}
a & b \\
c & d \\
\end{psmallmatrix}
\quad
\begin{pmatrix}
a & b \\
c & d \\
\end{pmatrix} \]
```

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix}$$

# Esetszétválasztás I

- esetszétválasztás: cases környezet – amsmath csomag

- $\left[ |x| = \right.$

- `\begin{cases}`

- `x, & \text{ha } x \geq 0, \\`

- `-x, & \text{ha } x < 0 \\ \end{cases}`

- `\quad \text{abszolútérték függvény} \\`

$$|x| = \begin{cases} x, & \text{ha } x \geq 0, \\ -x, & \text{ha } x < 0. \end{cases} \quad \text{abszolútérték függvény}$$

- táblázathoz hasonló szintaxis: cellák közé &, sor végére \\
- !! csak (max) két oszlop: formula és feltétel
- sorok száma tetszőleges!
- környező formulákkal egy sorba kerül (ha nem törjük meg kézzel), függőlegesen középre igazodik

# Esetszétválasztás II

- variánsok mathtools csomaggal:
- rcases: kapcsos zárójel (csak) jobb oldalon – pl. egyenletrendszerek összekapcsolására
- bal oldali formula oszlop alapértelmezésben `\textstyle` stílusban
- dcases: formula oszlop `\displaystyle` stílusban
- drcases: fentiek kombinációja, formula oszlop `\displaystyle` és zárójel jobb oldalon
- mindnek \*-os verziója, `cases*` stb.: a jobb oldali feltétel oszlop *szöveg módban* szedve
  - azaz nem kell a `\text{}` parancs bele
  - de ha formula is lenne benne, tegyük inline (sorközi) matematikai módba, `$ $` vagy `\( \)` közé

# Kapocs, magyarázat...

- mathtools csomaggal használjuk, jobb térköz
- kapocs formula alá
  - $\$ \backslash \text{underbrace}\{abc\} \$$   $\underbrace{abc}$
  - szöveg elhelyezése a kapocs alá: „alsó index”

$\$ \backslash \text{underbrace}\{abc\}_{=2} \$$

$\underbrace{abc}_{=2}$

- kapocs formula fölé:
  - $\$ \backslash \text{overbrace}\{abc\} \$$   $\overbrace{abc}$
  - szöveg a felső kapocs fölé: felső index

$\$ \backslash \text{overbrace}\{abc\}^{\{=2\}} \$$

$\overbrace{abc}^{\{=2\}}$

- analóg módon  $\backslash \text{underbracket}$  és  $\backslash \text{overbracket}$ , „szögletes zárójel” formájú kapocs mathtools csomaggal

# Relációs jel „dekoráció”

- kiegészítés relációs jel fölé és alá – amsmath csomaggal
  - !! argumentumok sorrendje nem intuitív, figyeljünk rá!
- fölé: `\overset`
  - $\overset{d}{\longrightarrow}$
- alá: `\underset`
  - $\underset{n \rightarrow \infty}{\longrightarrow}$
- fölé és alá
  - $\overset{d}{\underset{n \rightarrow \infty}{\longrightarrow}}$

- 5 Formulák számítása, igazítása
  - Számítás
  - Többsoros formulák típusai
  - Egyenletcsoport
    - Igazítatlan egyenletcsoport – `gather`
    - Igazított egyenletcsoport – `align`
    - Több oszlopos igazítás
  - Alszámítás
  - Egy egyenlet több sorban
  - Térköz hackelés

# Számozás

- emlékeztető: `equation` környezet, kb. a kiemelt matematikai mód `\[ \]` számozott verziója
  - számoz és matematikai módba vált
  - azaz, a `\begin{equation}`, `\end{equation}` *felváltják* a `\[`, `\]` parancsokat!
- alapértelmezés: formulák számozása folyamatos, szakaszcímeiktől független
  - `amsmath` csomaggal számozás pl. `section`-önként, `section`-ön belül alszámmal: `\numberwithin{equation}{section}`
- alapértelmezés: kiemelt formulák középre zárva
  - balra zárás `fleqn documentclass` opcióval (FLushleft EQUatioN)
- alapértelmezés: formulák sorszáma jobb oldalon
  - bal oldalon `leqno documentclass` opcióval (Left EQUation Numbers)

# Többsoros formulák típusai I

- alapértelmezésben kiemelt matematikai módban és equation-ben nem tehető sortörés!
- egyéb környezetekre lesz szükség
- két megközelítés:
  - **egyenletcsoport**, equation-t és matematikai módot **felváltó** környezetek, ahol minden sor külön sorszámot kap
    - **logikailag minden sor külön egyenlet**
    - pl. egy egyenletrendszer
  - matematikai módba (vagy azt felváltó környezetbe) **beágyazható** környezetek, ami (ha van sorszám,) közös sorszámmal tesz sortörést
    - **logikailag egyetlen összetartozó egyenlet**, ami szimplán túl hosszú hogy kiferjen egy sorba
    - pl. egy hosszú Taylor-sorfejtés, szita-formula, stb.



# Többsoros formulák típusai II

- közös parancsok:
  - sortörés, új egyenlet: `\\`
  - igazítási pontok kijelölése (ahol releváns): `&`
  - adott sorban egyenletsorszám elhagyása: `\nonumber` vagy `\notag` parancs
  - `\label`-t egy-egy számozott sornak érdemes adni
    - `amsmath` csomaggal `\eqref` parancs: tulajdonképpen (`\ref{...}`)
    - csak formázásban különbözik, bármilyen objektumtípusra használható
  - szöveg beszúrása egyenletek közé külön sorba: `\intertext`
    - miért jobb, mint lezárni a környezetet és simán szöveget írni?  
térközkezelés + igazítás megmarad!

## Igazítatlan egyenletcsoport I

- amsmath csomaggal gather környezet
- ha igazítatlan, miért jobb, mint több equation egymás után?  
a térközöket jobban kezeli!
- sortörés `\\`, számozatlan sor `\nonumber`

- `\begin{gather}`

$$a^2 + b^2 = c^2 \\$$

$$1 + 2 + 3 + 4 + 5 + 6 + 7 + 8 + 9 = 45$$

`\end{gather}`

$$a^2 + b^2 = c^2 \tag{1}$$

$$1 + 2 + 3 + 4 + 5 + 6 + 7 + 8 + 9 = 45 \tag{2}$$

## Igazítatlan egyenletcsoport II

- ezzel szemben

```
\begin{equation}
```

$$a^2 + b^2 = c^2$$

```
\end{equation}
```

```
\begin{equation}
```

$$1 + 2 + 3 + 4 + 5 + 6 + 7 + 8 + 9 = 45$$

```
\end{equation}
```

$$a^2 + b^2 = c^2 \tag{3}$$

$$1 + 2 + 3 + 4 + 5 + 6 + 7 + 8 + 9 = 45 \tag{4}$$

- `gather*` variáns: számozatlan – ismételt `\[ \]`-hez képest szintén a jobb térközkezelés az előnye!

## Igazított egyenletcsoport

- amsmath csomaggal align környezet
- sortörés `\\`, számozatlan sor `\nonumber`
- igazítási helyek `&` – tipikusan relációs jel (vagy műveleti jel) elé kerül

- `\begin{align}`

```
a^2 + b^2 &= c^2 \\
```

```
1 + 2 + 3 + 4 + 5 + 6 + 7 + 8 + 9 &= 45
```

```
\end{align}
```

$$a^2 + b^2 = c^2 \quad (5)$$

$$1 + 2 + 3 + 4 + 5 + 6 + 7 + 8 + 9 = 45 \quad (6)$$

- számozatlan verzió: `align*`

## Többoszlopos igazítás – align |

- align használható több oszlopos igazításra is!
- új sor `\\`, számozatlan sor `\nonumber`
- `&` **oszlop igazítása** és **oszlopelválasztó** is (!)
- oszlopok közt egyenletesen osztja a térközt
- `\begin{align}`

```
(f_1)'_x &= \cdots & (f_1)'_y &= \cdots &
```

```
(f_1)'_z &= \cdots \\
```

```
(f_2)'_x &= \cdots & (f_2)'_y &= \cdots &
```

```
(f_2)'_z &= \cdots
```

```
\end{align}
```

$$(f_1)'_x = \cdots \quad (f_1)'_y = \cdots \quad (f_1)'_z = \cdots \quad (7)$$

$$(f_2)'_x = \cdots \quad (f_2)'_y = \cdots \quad (f_2)'_z = \cdots \quad (8)$$

## Több oszlopos igazítás – flalign

- flalign környezet (flush align) – szintén amsmath csomagból
- szintaxis és működés megegyezik az előzővel – egyetlen különbség, hogy a szélső oszlopokat a margóhoz igazítja
- számozatlan flalign\* variáns

• `\begin{flalign}`

```
(f_1)'_x &= \cdots & (f_1)'_y &= \cdots &
```

```
(f_1)'_z &= \cdots & \\\
```

```
(f_2)'_x &= \cdots & (f_2)'_y &= \cdots &
```

```
(f_2)'_z &= \cdots
```

```
\end{flalign}
```

$$(f_1)'_x = \dots \qquad (f_1)'_y = \dots \qquad (f_1)'_z = \dots \quad (9)$$

$$(f_2)'_x = \dots \qquad (f_2)'_y = \dots \qquad (f_2)'_z = \dots \quad (10)$$

# Több oszlopos igazítás – alignat I

- alignat környezet (amsmath csomagból)
- használható a fentiekhez hasonlóan több oszlopos igazításhoz
- de az oszloptávolságot itt kézzel kell megadni(!)
- mivel nincs automatikus térköz, lineáris egyenletrendszer szépen rendezhető!
- új sor `\\`, számozatlan sor `\nonumber`
- oszlopok száma *kötelező argumentum*, pl. `\begin{alignat}{4}`
- & **oszlop igazítása** és **oszlopelválasztó** is(!)
- 4 oszlop esetén 4 **igazító** & jel és 3 **elválasztó** & jel (!)
- térköz opcionálisan az **elválasztó** & jel után illeszthető be – elég az első sorba

## Több oszlopos igazítás – alignat II

- használat lineáris egyenletrendszerhez

```
\begin{alignat}{4}
&+ x_1 &&+ 3 x_2 &&- x_3 &&= 5 \\
&+ 2 x_1 &&- x_2 &&&&= 2 \\
&- x_1 &&&&- 5x_3 &&= 0
\end{alignat}
```

$$+ x_1 + 3x_2 - x_3 = 5 \quad (11)$$

$$+ 2x_1 - x_2 = 2 \quad (12)$$

$$- x_1 - 5x_3 = 0 \quad (13)$$



## Több oszlopos igazítás – alignat III

- használat több oszlopos elrendezésre

```

\begin{alignat}{3}
(f_1)'_x &= \cdots & \&\quad (f_1)'_y &= \cdots \\
&\&\quad (f_1)'_z &= \cdots & \\
(f_2)'_x &= \cdots & \& (f_2)'_y &= \cdots \\
&\& (f_2)'_z &= \cdots & \\
\end{alignat}

```

$$(f_1)'_x = \cdots \quad (f_1)'_y = \cdots \quad (f_1)'_z = \cdots \quad (14)$$

$$(f_2)'_x = \cdots \quad (f_2)'_y = \cdots \quad (f_2)'_z = \cdots \quad (15)$$

# Alszámozás I

- cél: egy egyenletszámon belül betűzés, subfigure-höz hasonlóan
- amsmath csomagból subequations környezet
- szintaxis épp ellenkező a subfigure-rel: subequations-t az érintett számozott környezet(ek)en **kívül** kell nyitni és zárni(!)
  - kerülhet bele több számozott matematikai környezet és egyenletrendszer
  - kerülhet közé szöveg is
  - példa a következő dián!
- \label kerülhet
  - a subequations környezetbe – közös szám
  - következő oldali példával: `\eqref{eq:Pitag}` (16)
  - az egyes számozott sorokba – betűvel együtt
  - példánkban `\eqref{subeq:Pitag}` (16a)  
`\eqref{subeq:Pitag_trigon}` (16b)

## Alszámozás II

- ```
\begin{subequations}\label{eq: Pitag}
\begin{equation}\label{subeq: Pitag}
a^2 + b^2 = c^2
\end{equation}

```

És trigonometrikus formája

```
\begin{equation}\label{subeq: Pitag_trigon}
\sin^2 x + \cos^2 x = 1
\end{equation}
\end{subequations}
```

$$a^2 + b^2 = c^2 \tag{16a}$$

És trigonometrikus formája

$$\sin^2 x + \cos^2 x = 1 \tag{16b}$$

Egy egyenlet több sorban I

- emlékeztető: ld. 2. szakasz
- probléma: egyetlen, összetartozó egyenlet, ami nem fér ki egy sorba
- egyetlen sorszámmal – akár egy egyenletrendszer része
- megoldás: tördelés **beágyazható** környezettel
- beágyazható = ami matematikai módon vagy sorszámozott környezetben *belül* használható!
- a sorszámot függőlegesen középre igazítja

Egy egyenlet több sorban II

- példaul

```

\begin{align}
f(x) &= \cdots \\
\begin{split}
f(x) &\approx f(x_0) + f'(x_0) \cdot (x-x_0) \\
&+ \frac{1}{2} f''(x_0) \cdot (x-x_0)^2 \\
&+ \frac{1}{6} f'''(x_0) \cdot (x-x_0)^3 + \cdots
\end{split}
\end{align}

```

$$f(x) = \dots \tag{17}$$

$$f(x) \approx f(x_0) + f'(x_0) \cdot (x - x_0) + \frac{1}{2} f''(x_0) \cdot (x - x_0)^2 + \frac{1}{6} f'''(x_0) \cdot (x - x_0)^3 + \dots \tag{18}$$

Egy egyenlet több sorban III

- `split` – `amsmath` csomag
 - sortörés `\\` paranccsal, igazítás & jellel
 - a teljes sorszélességet kitölti – ne kerüljön mellé semmi!
 - nem error, csak kilóg a margóra, és `overfull box warning`
 - csak egyetlen & használható benne soronként
 - cserébe: a `split`-en belüli & jeleket nem csak egymáshoz igazítja, hanem a többi egyenlet & jeléhez is!
 - már ha van értelme, pl. ha `align-on` belül használjuk
 - a fenti példában ezért került egymás alá nem csak a `split`-en belül az \approx és $+$, hanem a másik egyenlet $=$ jele is!

Egy egyenlet több sorban IV

- gathered, aligned, alignedat – amsmath csomag
 - a gather, align, aligned beágyazható megfelelői
 - ennek megfelelő a belső szintaxis és viselkedés
 - szélességük a tartalomhoz igazodik
 - a cases-hez hasonlóan kerülhet vele egy sorba egyéb tartalom, zárójelezhető az egész struktúra, stb.
 - függőlegesen középre igazodik
 - de, kívülről nézve a tartalmuk csak egy *fekete doboz*(!)
 - azaz a belső & jeleket csak egymáshoz igazítja, a többi egyenlethez nem

Térköz hackelés

hasznos parancsok:

- `\phantom`: „láthatatlanul szedett szöveg” – igazításhoz, helykitöltéshez
- `\vphantom`, `\hphantom`: függőlegesen, ill. vízszintesen felveszi a tartalom kiterjedését, a másik irányban viszont 0 a mérete – láthatatlan helykitöltés, méretezés hackeléséhez
- `\mathstrut`: tulajdonképpen `\vphantom{()}`, gyorsabb használathoz
- vesd össze:
 - $\bar{a} + \bar{b}$
 - $\bar{a} + \bar{b}$