

# **L<sup>A</sup>T<sub>E</sub>X workshop s Vítkem v2.0**

## **„Studenti studentům“ na FEL ČVUT**

Vít Valečka

21. listopadu 2019

*„Stejně jako neexistuje ten nejlepší univerzální programovací jazyk, neexistuje ani ten nejlepší univerzální textový procesor. Podobně jako program pro Arduino nebudete psát v Javě a nativní aplikaci pro Android v C++, nedává smysl psát román v L<sup>A</sup>T<sub>E</sub>Xu a odbornou práci se spoustou matematických vzorců ve Wordu.“*

## Slovo úvodem

Tento dokument, byť je součástí krátkého workshopu s tématem publikačního systému  $\text{\LaTeX}$ , v žádném případě není kompletním popisem dovedností tohoto značkovacího jazyka, ani by neměl sloužit jako nějaká učebnice.

Cílem této práce je ukázat naprosté základy práce, upozornit na možná úskalí a navnadit čtenáře/posluchače k dalšímu studiu této problematiky, ať už samostatnému, nebo například prostřednictvím mezifakultního zápisu předmětu *01PSL Publikační systém  $\text{\LaTeX}$*  vyučovaném na Fakultě jaderné a fyzikálně inženýrské (FJFI) ČVUT v Praze, který vyučuje doktor Petr Ambrož.

# Obsah

<b>1</b>	<b>Trocha teorie</b>	<b>5</b>
<b>2</b>	<b>Základní struktura dokumentu</b>	<b>6</b>
2.1	Úvod do struktury . . . . .	6
2.2	Části dokumentu . . . . .	6
<b>3</b>	<b>Načítání rozšiřujících balíčků</b>	<b>7</b>
<b>4</b>	<b>Čeština v <math>\text{\LaTeX}</math>u</b>	<b>8</b>
<b>5</b>	<b>Formátování textu</b>	<b>9</b>
5.1	Znaky a formát textu . . . . .	9
5.2	Tok textu . . . . .	9
5.3	Rozložení stránky . . . . .	10
<b>6</b>	<b>Kapitoly</b>	<b>12</b>
<b>7</b>	<b>Seznamy</b>	<b>13</b>
7.1	Bodové odrážky . . . . .	13
7.2	Číslované odrážky . . . . .	13
7.3	Pojmenované odrážky . . . . .	13
<b>8</b>	<b>Tabulky</b>	<b>14</b>
8.1	Formátování tabulky . . . . .	14
8.2	Formátování sloupců . . . . .	14
<b>9</b>	<b>Matematika</b>	<b>17</b>
9.1	Základní pravidla . . . . .	17

9.2	Oddíly pro matematický zápis . . . . .	17
9.3	Znaky . . . . .	18
9.4	Mezery . . . . .	18
9.5	Zápisy . . . . .	18
<b>10</b>	<b>Obrázky</b>	<b>21</b>
<b>11</b>	<b>Reference, poznámky, odkazy</b>	<b>23</b>
11.1	Reference . . . . .	23
11.2	Poznámka pod čarou . . . . .	23
11.3	Použité zdroje . . . . .	23
11.3.1	Základní reference . . . . .	23
11.3.2	BibT <sub>E</sub> X . . . . .	24
<b>12</b>	<b>Vlastní příkazy</b>	<b>26</b>

## 1 Trocha teorie

$\text{\TeX}$  je program pro počítačovou sazbu. Vytvořil jej profesor Donald Ervin Knuth, který v 70. letech 20. století nebyl spokojen s tím, jak školní nakladatelství sázelo jeho skripta určená studentům. Ve skriptech se objevovalo mnoho chyb, především v matematických vzorcích, a ani typografie za mnoho nestála.

V současnosti je  $\text{\TeX}$  velmi populární zejména v akademických kruzích, zvláště v oborech jako je matematika, fyzika a informatika. Uplatnění ale může najít i v jiných oborech, např. humanitních – sazba je kvalitní nejen v případě matematiky, ale také v případě znaků různých abeced. Tímto programem byl z větší části vytlačen troff, formátovací systém preferovaný na mnoha Unixech. K dispozici je mnoho rozšiřujících balíků od velmi početných komunit uživatelů. Například existují balíky pro sazbu textu v podobě čárového kódu nebo vysazení obrázku situace šachové partie pomocí jediného příkazu. Taktéž existují komerční programové balíky.

$\text{\TeX}$  je obecně považován za nejlepší nástroj pro sazbu složitých matematických vzorců. Je však hojně používán i v běžné sazbě, stejně tak jako jeho odvozeniny a nadstavby, zvláště pak balík maker  $\text{\LaTeX}$ . I Wikipedie používá na svých stránkách tento systém pro matematické výrazy.

Zajímavostí je způsob označování verzí programu: místo tradičního zvyšování čísla verze se označení verze  $\text{\TeX}$  u prodlužuje o další číslici desetinného rozvoje Ludolfova čísla. Za nalezení chyby v programu nabízí Donald Knuth odměnu, která v současné době činí \$327,68. Další zajímavostí je skutečnost, že Donald Knuth vyjádřil přání, ať bezprostředně po jeho smrti přestane být  $\text{\TeX}$  dále vyvíjen a verze aktuální v toto datum se stane verzí finální, označenou právě číslem  $\pi$ .

$\text{\LaTeX}$  (vyslovuje se [latech]), je balík maker programu  $\text{\TeX}$ , který umožňuje autorům textů sázet a tisknout svá díla ve velmi vysoké typografické kvalitě, přičemž autor používá profesionály předdefinovaných vzhledů dokumentu.  $\text{\LaTeX}$  byl původně napsán Leslie Lamportem a užívá programu  $\text{\TeX}$  jako sázecího stroje.

Od 90. let je  $\text{\LaTeX}$  rozšiřován týmem  $\text{\LaTeX}3$ , který vede Frank Mittelbach. Tento tým se snaží sjednotit všechny rozšiřující verze  $\text{\LaTeX}u$ , které postupně vznikaly od verze 2.09. Aby byla nějakým způsobem odlišena stará verze od nové, byla zatím aktuální označena jako  $\text{\LaTeX}2\epsilon$ .

## 2 Základní struktura dokumentu

### 2.1 Úvod do struktury

Dokument psaný v  $\text{\LaTeX}$  se dělí na dvě základní části – *preambuli* a *tělo dokumentu*.

**Preamble** obsahuje importy používaných balíčků, nastavení dokumentu (např. velikosti okrajů stránky, styly odstavců a textu atd.), informace o dokumentu samotném, definice vlastních příkazů a další.

**Tělo dokumentu** pak zahrnuje samotný obsah, který chceme sdělit. Patří sem tedy samotný text, kapitoly, obrázky, tabulky atd.

Při psaní dokumentu je také dobré myslet na 3 základní vlastnosti:

- libovolný počet mezer v textu se bere jako jedna mezera
- mezery na začátku řádku jsou ignorovány
- prázdný řádek (popř. libovolný počet prázdných řádků) se bere jako nový odstavec

V  $\text{\LaTeX}$  také platí, že:

- všechny příkazy začínají  $\backslash$  (zpětným lomítkem)
- příkaz může mít hlavní a vedlejší parametry – `prikaz[vedlejsi]{hlavni}`
- speciální znaky se „escapují“  $\backslash$  (zpětným lomítkem)

### 2.2 Části dokumentu

Každý  $\text{\LaTeX}$ ový dokument obsahuje tři základní značky:

`documentclass{class}` – určuje typ dokumentu (jedná se v podstatě o „šablonu“ pro vzhled). Za parametr „class“ se doplňuje název šablony. Mezi nejpoužívanější třídy dokumentu patří **article**, **book** a **report**.

`begin{document}` – značí začátek samotného dokumentu – zde se vše zapisuje.

`end{document}` – značí konec samotného dokumentu – vše za touto značkou se ignoruje.

### Příklad 1 – Hello World!

Kód níže vytvoří dokument, jehož jediným obsahem bude text „Hello World!“.

```
\documentclass{article}
\begin{document}
  Hello World!
\end{document}
```

## 3 Načítání rozšiřujících balíčků

Načítání balíčků, které rozšiřují základní funkcionalitu se provádí v preambuli pomocí příkazu `\usepackage[]{}{}`.

Do prvních (hranatých) závorek se uvádí volitelné parametry, do druhých (složených) povinné parametry - typicky název balíčku, který načítáme.

## 4 Čeština v L<sup>A</sup>T<sub>E</sub>Xu

Základní font, který je využíván, není přizpůsoben pro korektní zobrazení českého jazyka – konkrétně pak diakritiky. Z tohoto důvodu pro správnou sazbu využíváme balíček *Babel* a některou ze sad českých písem, typicky *fontenc*.

Pro korektní zobrazení speciálních znaků a češtiny je také dobré přepnout kódování dokumentu do UTF-8 popřípadě CHCP 1250.

`\usepackage[czech]{babel}` – načte balíček obsahující pravidla pro češtinu (formáty data, odstavců atd.).

`\usepackage[utf8]{inputenc}` – nastaví dokument, aby využíval pro kódování textu UTF-8.

`\usepackage[IL2]{fontenc}` – načte balíček s českým fontem.

### Příklad 2 – Žluťoučký kůň

Kód níže vytvoří dokument, jehož jediným obsahem bude text „*Příliš žluťoučký kůň úpěl ďábelské ódy.*“.

```
\documentclass{article}
\usepackage[czech]{babel}
\usepackage[utf8]{inputenc}
\usepackage[IL2]{fontenc}
\begin{document}
  Příliš žluťoučký kůň úpěl ďábelské ódy.
\end{document}
```



## 5 Formátování textu

Většina atributů formátování dokumentu a v něm obsaženého textu záleží na využití šablony. Některé možnosti jsou proto v rámci L<sup>A</sup>T<sub>E</sub>Xu omezené.

Dále následuje seznam základních značek pro formátování textu.

### 5.1 Znaky a formát textu

`%` – jednořádková poznámka v kódu (L<sup>A</sup>T<sub>E</sub>X nemá víceřádkové poznámky)

`-` `(-)` – spojovník

`--` `(—)` – pomlčka

`---` `(—)` – dlouhá pomlčka

`emph{text}` (*text*) – zvýraznění textu (s ohledem na aktuálně použitou šablonu)

`textbf{text}` (**text**) – tučný text

`textit{text}` (*text*) – text psaný kurzívou

`textrm{text}` – normální text (pokud například styl dokumentu zavádí, že standardně je text kurzívou)

`underline{text}` (text) – podtržený text

`texttt{aimnl}` (`aimnl`) – neproporcionální text

`textsc{text}` (TEXT) – text v kapitálkách

`textsf{text}` (**text**) – bezpatkový text

`indent` – odsazení prvního řádku odstavce dle předpisů v šabloně, nebo uživatelem definovaných

`noindent` `(-)` – zruší odsazení prvního řádku odstavce

`uv{text}` („text“) – umístí text do uvozovek dle aktuálně použité jazykové sady

### 5.2 Tok textu

`\` – přechod na nový řádek

`newpage` – vloží konec stránky

`clearpage` – vloží konec stránky

`hspace{2em}` – vloží horizontální mezeru do textu (možno využít všechny obvyklé jednotky – px, em, cm, mm...)

`vspace{2em}` – vloží vertikální mezeru

`vspace*{2em}` – vloží vertikální mezeru mezi „něco“ a „nic“ (např. mezi text a okraj stránky)

`mbox{text}` – obaluje text, který nesmí být rozdělený na konci řádku

`begin{flushleft} ... end{flushleft}` – zarovnání textu dle zadaného parametru – existují tři možnosti `flushleft` – vlevo, `flushright` – vpravo a `center` – na střed.

### 5.3 Rozložení stránky

Studium tohoto tématu ponecháme na čtenáři. Omezíme se pouze na informaci, že aktuální rozložení stránky lze vypsát příkazem `layout` (je součástí stejnojmenného balíčku), zde jsou vypsány i příkazy, kterými lze jednotlivé parametry (okraje, šířku prostoru pro text...). Jako konkrétní příklad můžeme uvést např.:

```
setlength{\textheight}{24.7cm}
```

První parametr určuje, kterou vlastnost chceme měnit, druhý pak nastavovanou velikost.

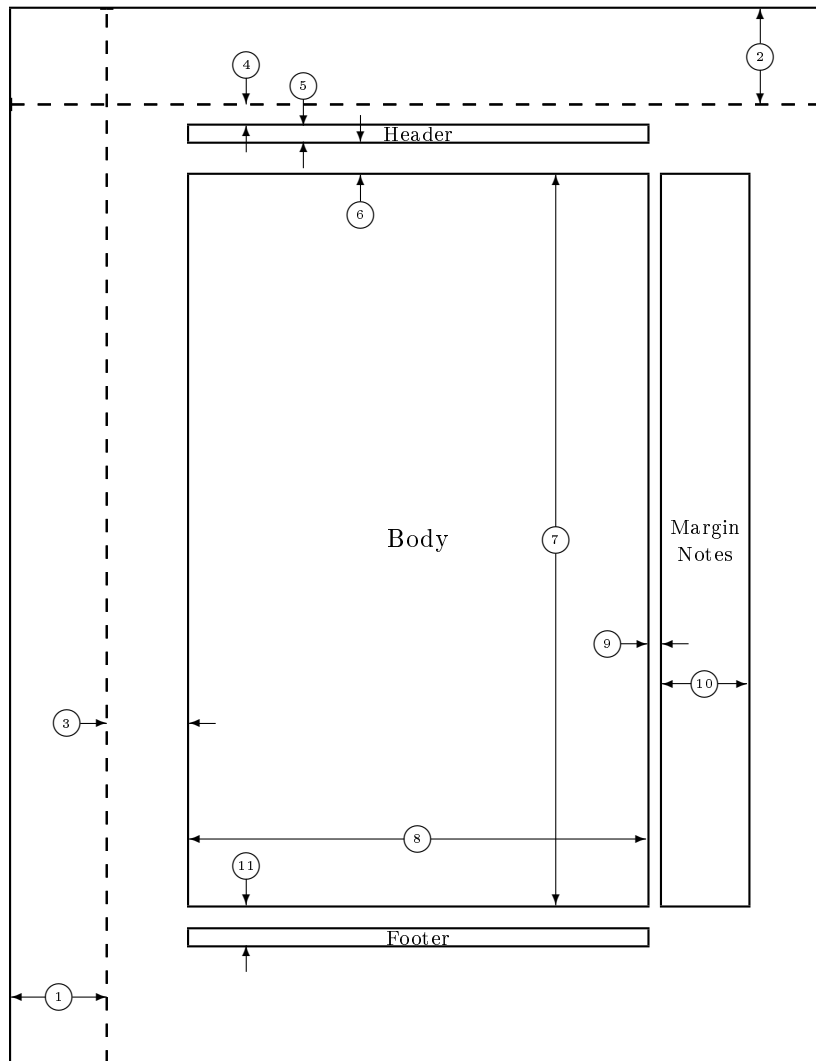
Lze také volit nastavení jen pro sudé nebo liché stránky:

```
setlength{\oddsidemargin}{-0.04cm}
```

A je také možné pracovat s jednotlivými již nastavenými parametry a provádět s nimi jednoduché matematické operace (sčítání, odčítání):

```
setlength{\topmargin}{-\headheight-\headsep}
```

Následující stránka obsahuje náhled rozložení tohoto dokumentu. Jedná se pouze o mírně upravenou šablonu `article`.



1	one inch + \hoffset	2	one inch + \voffset
3	\oddsidemargin = 62pt	4	\topmargin = 16pt
5	\headheight = 12pt	6	\headsep = 25pt
7	\textheight = 550pt	8	\textwidth = 345pt
9	\marginparsep = 11pt	10	\marginparwidth = 65pt
11	\footskip = 30pt		\marginparpush = 5pt (not shown)
	\hoffset = 0pt		\voffset = 0pt
	\paperwidth = 614pt		\paperheight = 794pt

## 6 Kapitoly

`section{nadpis}` – vloží číslovaný nadpis kapitoly

`subsection{nadpis}` – vloží číslovaný nadpis podkapitoly, přidáním dalších předpon „sub“ lze více vnořovat

`section*{nadpis}` – vloží nečíslovaný nadpis kapitoly, analogicky funguje také pro podkapitoly

Veškeré použité kapitoly lze složit do seznamu, který je možné v libovolné části dokumentu vypsat ve formě obsahu příkazem `tableofcontents{}`. Jak podobný obsah vypadá se může čtenář přesvědčit v úvodu tohoto dokumentu.

## 7 Seznamy

### 7.1 Bodové odrážky

#### Příklad 3

<code>\begin{itemize}</code>	• semper vel quam at
<code>\item semper vel quam at</code>	
<code>\item rutrum tristique metus</code>	• rutrum tristique metus
<code>\item[:)] vivamus elementum lectus</code>	:) vivamus elementum lectus
<code>\end{itemize}</code>	

### 7.2 Číslované odrážky

#### Příklad 4

<code>\begin{enumerate}</code>	1. donec non hendrerit erat
<code>\item donec non hendrerit erat</code>	
<code>\item et suscipit turpis</code>	2. et suscipit turpis
<code>\item donec interdum turpis</code>	
<code>\end{enumerate}</code>	3. donec interdum turpis

### 7.3 Pojmenované odrážky

#### Příklad 5

```
\begin{description}
\item[Suspendisse] non risus ante. Donec sed pharetra massa...
\item[Quisque] non augue ut libero vehicula ullamcorper at quis...
\end{description}
```

**Suspendisse** non risus ante. Donec sed pharetra massa. Nam leo libero, condimentum in sollicitudin eget, porta sed quam.

**Quisque** non augue ut libero vehicula ullamcorper at quis mauris. Phasellus tincidunt velit eget dui malesuada ultrices.

## 8 Tabulky

Tvorba tabulek je v  $\text{\LaTeX}$ u podobná tvorbě v HTML. Bez složitého úvodu si vše ukážeme na příkladu:

### Příklad 6

```
\begin{tabular}{|l|l|c|p{4cm}|}  
\hline Kdo & Kolik & Poznámka & \\  
\hline Alice & 151 & To je docela dost, otázka jak k tomu přišla. & \\  
Bob & 21 & Nic moc Bobe, zaber. & \\  
\hline  
\end{tabular}
```

Kdo	Kolik	Poznámka
Alice	151	To je docela dost, otázka jak k tomu přišla.
Bob	21	Nic moc Bobe, zaber.

### 8.1 Formátování tabulky

`\begin{tabular}{params}` – začátek definice tabulky, slovo `tabular` specifikuje, že se jedná o tabulku, ve druhých složených závorkách je pak specifikováno rozložení tabulky – sloupce, jejich ohraničení, zarovnání textu, popř. pevná šířka

`\end{tabular}` – konec tabulky

`\hline` – vytvoří vodorovné ohraničení mezi řádky tabulky, uvádí se na první řádek pod ohraničením

`&` – přechod na novou buňku

`\ \` – přechod na nový řádek

### 8.2 Formátování sloupců

`r` – sloupec zarovnaný vpravo

`l` – sloupec zarovnaný vlevo

`c` – sloupec zarovnaný na střed

`p{4cm}` – sloupec s pevnou šířkou (zde 4 cm)

| – oddělí vedle sebe ležící sloupce čarou (přidá mezi ně ohraničení)

@{+++} – definice vlastního oddělovače sloupců (zde tři znaky +), lze ale zvolit libovolný jiný znak, nebo řetězec znaků

### Příklad 7 - Zarovnání na desetinnou čárku (trik)

Alice	1,51
Bob	12,3

```
\begin{tabular}{lr@{,}l}
\hline Alice & 1 & 51 \\
Bob & 12 & 3 \\
\hline
\end{tabular}
```

### Příklad 8 - Alternativní ohraničení (Booktabs)

D	$P_u$	$u_u$	$\beta$	$G_f$
[in]	[lbs]	[in]		[psi·in]
5	269.8	0.000674	1.79	0.04089
10	421.0	0.001035	3.59	0.04089
20	640.2	0.001565	7.18	0.04089

```
\begin{tabular}{ccccc}
\toprule[1pt] D & $P_u$ & $u_u$ & $\beta$ & $G_f$ \\
{[in]} & [lbs] & [in] & & [psi·in] \\
\midrule[1pt] 5 & 269.8 & 0.000674 & 1.79 & 0.04089 \\
\midrule 10 & 421.0 & 0.001035 & 3.59 & 0.04089 \\
\midrule 20 & 640.2 & 0.001565 & 7.18 & 0.04089 \\
\bottomrule[1pt]
\end{tabular}
```

### Příklad 9 - Multirow, multicolumn

A	B	C	D	F					
1	0	6	230	35	40	55	25	40	35
	1	5	195	25	50	35	40	45	

```
\begin{tabular}{|c|c|c|c|p{1cm}p{1cm}p{1cm}p{1cm}p{1cm}p{1cm}|}
\hline
A & B & C & D & \multicolumn{6}{c}{F} \\ \hline
\multirow{2}{*}{1} & 0 & 6 & 230 & 35 & 40 & 55 & 25 & 40 & 35 \\
& 1 & 5 & 195 & 25 & 50 & 35 & 40 & 45 & \\ \hline
\end{tabular}
```

V případě příkazu `\multicolumn{7}{c}{text}` první parametr určuje počet sloupců, které „multisloupec“ zabírá, druhý určuje zarovnání (včetně případného ohraničení) a třetí pak obsah.

U příkazu `\multirow{2}{*}{text}` první parametr určuje počet zabíraných řádek (řádek uvedený příkazem je první), druhý šířku (lze samozřejmě také použít znak `*` – pak je šířka automatická) a poslední argument opět obsahuje text.

Za zmínku ještě stojí příkaz tzv. *dcolumn* – je to speciální typ sloupce, který umožňuje snadné zarovnávání podle určitého parametru – například konkrétního znaku. Při použití formulace `D{:}{:}{3.3}` dojde k zarovnání podle dvojtečky tak, že na každé straně bude místo na tři platné číselné pozice. První argument specifikuje, jaký je referenční znak pro zarovnání ve zdrojovém kódu, druhý určuje znak pro zarovnání ve výstupním PDF (je tedy možné použít jiný znak v kódu  $\LaTeX$ u a jiný si nechat vypisovat do PDF), třetí pak počet platných číslic, které se mají vypsát. Je také možné specifikovat jiný počet číslic pro každou stranu zarovnání.

### Příklad 10 - D-column

Id...Příjmení	Jméno	
1...Jaroš	Jan	4'5
2...Jančařík	Václav	6'01
3...Okruhlica	Štefan	33'3333

```
\begin{tabular}{|r@{...}l|p{3cm}D{,}{'}{3.4}|}
Id & Příjmení & Jméno & \\
\hline
1 & Jaroš & Jan & 4,5 \\
2 & Jančařík & Václav & 6,01 \\
3 & Okruhlica & Štefan & 33,3333 \\
\end{tabular}
```



## 9 Matematika

Matematický zápis nám v  $\text{\LaTeX}$  umožňuje balíček  $\text{AMSMath}$  projektu  $\mathcal{AMS}\text{-}\text{\LaTeX}$ .

### 9.1 Základní pravidla

Podobně jako u běžného textu, i matematický mód  $\text{\LaTeX}$  se řídí několika jednoduchými pravidly, která je při sázení dobré mít na paměti:

- mezery jsou ignorovány (libovolný počet mezer = 0 mezer)
- v sekci pro matematický zápis se nesmí objevit prázdný řádek
- výchozí styl písma je kurzívou
- každé písmeno  $\text{\LaTeX}$  považuje za samostatnou proměnnou
- v matematickém zápisu není možné užívat stejné příkazy, jako v běžném textu

### 9.2 Oddíly pro matematický zápis

Matematický zápis můžeme využívat ve dvou různých stylech. Jako součást odstavce souvislého textu – tzv. *in-line*, nebo jako samostatný blok zabírající celou šířku stránky – chová se jako *vlastní odstavce*.

Pro *in-line* zápis můžeme použít následující uvozující znaky:

- `\begin{math} ... \end{math}`
- `\( ... \)`
- `$ ... $`

Pro zápis do *vlastního odstavce* můžeme použít následující uvozující znaky:

- `\begin{displaymath} ... \end{displaymath}`
- `\[ ... \]`
- `$$ ... $$`
- `\begin{equation} ... \end{equation}`

### 9.3 Znaky

**mocnina/horní index**  $- x^2 - x^2$ , resp.  $x^{\{2\}}$

**dolní index**  $- x_2 - x_2$ , resp.  $x_{\{2\}}$

**odmocnina**  $- \sqrt{x}$ , popř.  $\sqrt[3]{x} - \sqrt[3]{x}$ , popř.  $\sqrt[3]{x}$

**zlomek**  $- \frac{a}{b} - \frac{a}{b}$

**suma**  $- \sum_{i=0}^n x^a - \sum_{i=0}^n x^a$  – zápis je v podstatě znak sumy s dolním a horním indexem

**integrál**  $- \int_{i=0}^n x^a - \int_{i=0}^n x^a$  – obdobně jako v případě sumy se jedná o znak integrálu s dolním a horním indexem

**produkt**  $- \prod_{i=0}^n x^a - \prod_{i=0}^n x^a$

**funkce**  $- \sin(x) \cos(x) \tan(x) \max_{x \in M} \{x\} - \sin(x) \cos(x) \tan(x) \max_{x \in M} \{x\}$   
– všechny základní funkce mají svůj vlastní příkaz

### 9.4 Mezery

Jak již bylo uvedeno, matematický zápis ignoruje všechny mezery. Pokud je tedy chceme do zápisu přidat,  $\mathcal{A}\mathcal{M}\mathcal{S}\text{-}\mathcal{L}\mathcal{A}\mathcal{T}\mathcal{E}\mathcal{X}$  nám poskytuje několik možností:

- $\quad$  – mezera šířky písmene M
- $\quad$  – 1/2 quadu
- $\quad$  – 2 quady
- $\quad$  – 3/18 quadu
- $\quad$  – 4/18 quadu
- $\quad$  – 5/18 quadu

### 9.5 Zápisy

Matematický zápis je v  $\mathcal{L}\mathcal{A}\mathcal{T}\mathcal{E}\mathcal{X}$ u na pokročilé úrovni. Často však uvnitř vlastního odstavce s rovnicemi chceme zapsat i prostý text. To nám umožňuje značka  $\text{\texttt{\text{abc}}}$ :

$$x^2 \geq 0 \text{ pro všechna } x$$

Problém může nastat při zápisu závorek. Ty by velikostně měly odpovídat velikosti obsahu. Z tohoto důvodu nemůžeme používat standardní znaky, ale místo nich speciální značky `\left(` a `\right)` (analogicky pro hranaté závorky):

$$\left(\frac{a}{b}\right)^2 - \left[\frac{a}{b}\right]^2$$

Často v matematice také používáme tři tečky jako výpustku. I ta má v  $\text{\LaTeX}$  svou vlastní značku. Je však třeba rozlišovat, zda potřebujeme tečky na úrovni základní linky textu (např. prvky v seznamu – použijeme `\ldots`) nebo na středu řádku (např. výpustka dlouhého seznamu prvků k součtu – použijeme `\cdots`).  $\text{\LaTeX}$  také poskytuje do určité míry „chatré“ řešení v podobě značky `\dots`, ta by měla z kontextu (okolního textu) využít správný druh teček (ukázka v druhém řádku příkladu níže).

$$\begin{array}{l} i \in \{1, 2, \dots, n\} \quad x_1 + x_2 + \cdots + x_n \\ i \in \{1, 2, \dots, n\} \quad x_1 + x_2 + \cdots + x_n \end{array}$$

V neposlední řadě je dobré myslet také na zápis matic – ten je velmi podobný tabulkám a často se zde využívá v předešlé kapitole zmíněný `dcolumn` (kvůli zarovnání na desetinnou čárku). Příklad si můžete prohlédnout níže:

$$A = \begin{pmatrix} 1 & -24.5 \\ 3 & 5.22 \end{pmatrix}$$

```

 $\$$ 
 $A=\left($ 
 $\begin{array}{@{}cD{.}{.}{.}{3.2}@{}}$ 
 $1 \& -24.5 \backslash$ 
 $3 \& 5.22$ 
 $\end{array}\right)$ 
 $\$$ 

```

Další zajímavost se týká značky `\begin{equation}`, výsledná rovnice je totiž číslována a jde se na ni odkazovat pomocí značky `\label{rov1}` a `\ref{rov1}`, jak můžete vidět v rovnici 1:

$$\mathcal{X} = \frac{1}{n} \sum_{i=1}^r p_i x_i \quad \mathcal{S} = \sqrt{\frac{1}{n} \sum_{i=1}^r p_i (x_i - \mathcal{X})^2}, \quad (1)$$

```

\begin{equation}
\label{xx}
\mathcal{X}=\frac{1}{n}\sum_{i=1}^rp_{ix_i} \quad
\mathcal{S}=\sqrt{\frac{1}{n}\sum_{i=1}^rp_i(x_i-\mathcal{X})^2},
\end{equation}

```

Rovnice lze také spojovat do skupin (zde například skupina rovnic 2 obsahující rovnice 2a a 2b):

$$x = a + b \tag{2a}$$

$$y = c - d + 15 \tag{2b}$$

```

\begin{subequations}\label{eq:x}
\begin{align}
x &= a + b \label{eq:x1} \\
y &= c - d + 15 \label{eq:x2}
\end{align}
\end{subequations}

```

A také lze místo běžného číslování rovnice rovnou pojmenovávat názvy. Mějte ale na paměti, že názvy neslouží jako reference:

$$\lim_{n \rightarrow \infty} \sqrt[n]{\left(\frac{n}{n+1}\right)^{n^2}} = \lim_{n \rightarrow \infty} \left(1 + \frac{1}{n}\right)^{-n} = \frac{1}{e} < 0 \tag{Cauchy}$$

```

\begin{gather}
\lim_{n \rightarrow \infty} \sqrt[n]{\left(\frac{n}{n+1}\right)^{n^2}} = \\
\lim_{n \rightarrow \infty} \left(1 + \frac{1}{n}\right)^{-n} = \\
\frac{1}{\mathrm{e}} < 0 \tag{Cauchy}
\end{gather}

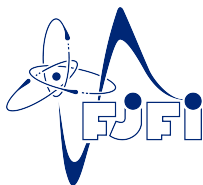
```

Často chceme také zapsat znak číselné množiny, jako jsou reálná čísla, přirozená čísla atd. K tomuto není nutný speciální příkaz, ale využití správného fontu z balíčku *AMSTeX*, který je k dispozici pouze v matematickém zápisu. Máme tedy reálná čísla –  $\mathbb{R}$  ( `$\mathbb{R}$` ), racionální čísla –  $\mathbb{Q}$  ( `$\mathbb{Q}$` ), celá čísla –  $\mathbb{Z}$  ( `$\mathbb{Z}$` ), přirozená čísla –  $\mathbb{N}$  ( `$\mathbb{N}$` ) a také komplexní čísla –  $\mathbb{C}$  ( `$\mathbb{C}$` ).

## 10 Obrázky

Vkládání obrázků je na první pohled složitější, než v případě např. Wordu. Jeho výsledek je také nejistý, protože o výsledném umístění obrázku nakonec rozhodne sám  $\text{\LaTeX}$ , a to bez vašeho vědomí.

Máme dvě možnosti, jak zaobalit obrázek pro vložení do dokumentu. Buď do značky `figure`, nebo `table`. Níže si ukážeme například první možnost:



Obrázek 1: Logo FJFI

```
\begin{figure}[!htp]
\begin{center}
\includegraphics[scale=0.1]{logo-fjfi.png}
\end{center}
\caption{Logo FJFI}
\label{fig:logo}
\end{figure}
```

U značky `figure`, první parametr určuje, kam se obrázek umístí. Možné hodnoty parametru jsou následující:

- `h` – here – umístění zde
- `t` – top – k hornímu okraji stránky
- `b` – bottom – k dolnímu okraji stránky
- `p` – page – samostatná stránka s obrázkem
- `!` – slouží k potlačení méně důležitých omezení pro umístění obrázku

Parametry lze samozřejmě libovolně kombinovat –  $\text{\LaTeX}$  se je bude snažit v uvedeném pořadí plnit. Pro zarovnání se často jako parametr využívá řetězec `!htp` – tedy „*potlač méně důležitá omezení a umísti obrázek přímo zde, nejde-li to tak k hornímu okraji stránky a pokud nejde ani to, pak na samostatnou stránku s obrázkem*“.

Zmínit musíme také `\includegraphics[scale=0.1]{logo-fjfi.png}`, který se stará o samotné načtení obrázku. První (volitelný) parametr udává násobící člen pro zvětšení/zmenšení obrázku, druhý (povinný) pak cestu k souboru s obrázkem.

Značka `caption` slouží jako viditelný popis obrázku, přes `label` se lze na obrázek standardně odkazovat – viz. Obrázek 1.

Chcete-li si na začátku/konci dokumentu nechat vytvořit seznam všech obrázků, je k tomu možné využít příkazy `\listoffigures` nebo `\listoftables`, které jej pro vás automaticky vygenerují.

## 11 Reference, poznámky, odkazy

### 11.1 Reference

L<sup>A</sup>T<sub>E</sub>X umožňuje snadné reference na stránky, nebo kapitoly. V případě *reference na kapitolu* stačí za příslušnou značku `section` doplnit `label{nazev}`, kde „nazev“ je zvolené jméno, přes které se pak budeme na referenci odvolávat. Kdekoliv v textu se pak můžeme odvolat na to, že v sekci 11.1 (`ref{reference}`) na stránce 23 (`pageref{reference}`) se něco nachází.

### 11.2 Poznámka pod čarou

Narozdíl například od Wordu se v L<sup>A</sup>T<sub>E</sub>Xu uvádí poznámka pod čarou přímo na místo, kde se nachází. Vždy se tak po vytvoření PDF bude nacházet na správné stránce<sup>1</sup>.

Pro její vytvoření se využívá příkaz `footnote{text}`.

### 11.3 Použité zdroje

#### 11.3.1 Základní reference

Jednoduchou správu může poskytovat i L<sup>A</sup>T<sub>E</sub>Xsám, není ale příliš efektivní a výstupy nemusí splňovat podmínky pro formát uvádění zdrojů. Viz. příklad níže:

#### Reference

- [1] Leslie Lamport, *L<sup>A</sup>T<sub>E</sub>X: a document preparation system*, Addison Wesley, Massachusetts, 2nd edition, 1994.

```
\begin{thebibliography}{9}
\bibitem{lamport94}
Leslie Lamport,
\textit{\LaTeX: a document preparation system},
Addison Wesley, Massachusetts,
2nd edition,
1994.
\end{thebibliography}
```

---

<sup>1</sup> Lorem ipsum dolor sit amet.

Je také možné se na jednotlivé zdroje odvolávat – [1, chapter, p. 215]. Nadpis je generován automaticky, jeho odstranění je bohužel komplikované.

### 11.3.2 BibTeX

Mnohem sofistikovanějším a robustnějším nástrojem pro správu zdrojů a citací je modul BibTeX. Má velmi obsáhlou dokumentaci, kterou naleznete na webu balíčku, i třeba na webu online sázecího prostředí Overleaf. Stručný a věcný návod v češtině má také na svém školním webu Ing. David Martinek z FIT VUT v Brně<sup>2</sup>.

Základem je načtení balíčku cite, který zprostředkovává veškeré funkce. Dále je třeba samostatný soubor BibTeXovou databází. Ten se musí nacházet ve stejné složce, jako \*.TEX soubor, ve kterém chceme BibTeX použít. V našem příkladu jej pojmenujme např. „citace.bib“ a jeho obsah může vypadat například takto:

```
@BOOK{Ceska1994,
author = "Milan ČEŠKA",
title = "Petriho Sítě",
year = "1994",
publisher = "Akademické nakladatelství CERM Brno",
isbn = "80-85867-35-4",
owner = "David Martinek"
}

@MISC{simlib,
author = "Petr PERINGER",
title = "SIMLIB/C++",
year = "Naposledy navštíveno 31. 3. 2005",
howpublished = "[online]",
url = "http://www.fit.vutbr.cz/~peringer/SIMLIB/"
}
```

Tuto databázi je třeba následně do dokumentu vložit – podobně a ve stejném místě jako v případě příkazu thebibliography z předešlé kapitoly:

```
\bibliographystyle{alpha}
\bibliography{citace}
```

V příkazu bibliographystyle{alpha} určujeme vzhled/formát, v jakém

---

<sup>2</sup>Web najdete na adrese <http://www.fit.vutbr.cz/~martinek/latex/bibtex.html>



se má citace vypsát do výsledného dokumentu. Mezi nejčastější parametry patří `alpha`, `plain`, `unsrt`, `abbrv` a `czechiso`.

Řetězec `bibliography{citace}` pak určuje název souboru `*.BIB`, ve kterém má `LaTeX` hledat databázi pro `BibTeX`.

Takovéto citace pak přímo v textu vypadají takto[1], popřípadě takto[2] a následný výpis referencí si můžete prohlédnout níže.

## Reference

- [1] Milan ČEŠKA. *Petriho Síť*. Akademické nakladatelství CERM Brno, 1994.
- [2] Petr PERINGER.

Při práci s `BibTeXem` je třeba myslet na to, že aby se změny projevíly, může být třeba více než jedno sestavení dokumentu, popřípadě může být pro zobrazení změn nutné zvlášť sestavit nejprve databázi `BibTeXu` a až následně samotný dokument.

## 12 Vlastní příkazy

L<sup>A</sup>T<sub>E</sub>X umožňuje definování vlastních příkazů. Je tak možné doplnit určitou funkcionalitu, která není snadno dostupná, popřípadě zjednodušit složité zápisy, které se často opakují. Lze tak dosáhnout snazšího zápisu, který je mnohem lépe čitelný.

Pro definování vlastního příkazu se využívá příkaz `newcommand`, který píšeme do preambule dokumentu. Je specifikován takto:

```
newcommand{\R}[1]{\mathbb{R}_{#1}}
```

V prvních složených závorkách je uvedena značka vlastního příkazu, kterou se na něj budeme odvolávat v dokumentu. V hranatých závorkách je uveden počet argumentů, které lze k příkazu přidružit (maximálně 9). V posledních složených závorkách je pak řetězec, který říká, co značka dělá (v tomto případě vypíše v matematickém zápisu  $\mathbb{R}$ , jako značku množiny reálných čísel). V definici příkazu se lze na jednotlivé parametry odvolávat přes #, za kterou je uvedeno číslo označující pořadí parametru (indexováno je od 1).

## Poděkování

Tento projekt a doprovodné materiály by nikdy nevznikly bez (ať už vědomého, nebo nevědomého) přispění několika lidí. Poděkování patří:

**Ing. Petru Ambrožovi, Ph.D.** za vedení předmětu *01PSL Publikační systém L<sup>A</sup>T<sub>E</sub>X* při FJFI ČVUT v Praze. Bez jeho nadšení a zápalu pro L<sup>A</sup>T<sub>E</sub>X by nevznikly tyto materiály, ani workshop.

**Bc. Beatě Karlové** za iniciativu a organizaci prvního ročníku. Bez ní by tento projekt nikdy nevznikl.

**Ing. Pavlu Náplavovi, Ph.D.** , který za CZM FEL ČVUT umožnil konání workshopů a organizoval druhý ročník. Díky němu došlo také ke zveřejnění materiálů na webu fakulty, takže je může využívat opravdu každý.