

TeX a L^ATeX

Vít Baisa

- Jeden ze zakladatelů počítačových věd (computer science),
- emeritní profesor FI MU (1996),
- autor *The Art of Computer Programming* (neustále pokračuje);
- nespokojen s kvalitou a rychlostí sazby (mnoho chyb v matematických vzorcích zanesených sazeči),
- vyvíjel deset let systém $\text{T}_{\text{E}}\text{X}$ → zdarma,
- a METAFONT.

- T_EX z řeckého $\tau\epsilon\chi$ (technologie a umění).
- Univerzálnost (různé abecedy, pravolevá sazba, sloupcová sazba, ...), přesnost,
- bezchybovost (Knuth vyplácí šeky za objevení chyby),
- nezávislost na platformě – UNIX, OS X, Windows aj.
- Řada knih *Computers & Typesetting*.
- T_EX šířen od vzniku (1983) zdarma, bez větších změn.
- Jedná se o makrojazyk, příkazy pro sazbu.
- Obsahuje 300 primitivních příkazů (`\vskip`, `\def`, `\noindent`, ...).
- Formáty plain, L^AT_EX.

- Tvůrce Leslie Lamport.
- Zpřístupnění poměrně složitého systému běžným uživatelům;
- ti neřeší *jak*, ale *co* chtějí sázet.
- Pro vysázení názvu podkapitoly se použije příkaz `\section{Úvod}`.
- Ten sám vybere písmo, velikost, odsazení, vložení do obsahu, zvýšení čísla podkapitoly atd.

- Zdrojový textový soubor (`.tex`),
- obsahuje samotný text pro sazbu a příkazy $\text{T}_{\text{E}}\text{X}$ u a $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ u,
- nesmí obsahovat žádné jiné speciální sekvence, zejména musí být tzv. *plaintextový*.
- Překladač ($\text{T}_{\text{E}}\text{X}$) určuje rozmístění jednotlivých prvků sazby (od písmen až po celé strany),
- sází na základě rozměrů, neví nic o tvarech výsledného písma.
- Překladač vytvoří soubor `.dvi` (DeVice Independent), z něj pak
- formát PostScript, PDF.

Pracovní postup:

- příprava zdrojového textu,
- překlad – vysázení a
- prohlížení.

Výhody:

- použití libovolného textového editoru,
- automatizovaná úprava textu a
- přenositelnost.

- aktivní znak: \$, %, & ,
- posloupnost \\, \% a
- posloupnost \slovo (\large, \Phi).
- T_EX rozlišuje malá a velká písmena v příkazech.

Je třeba rozlišovat mezi příkazy a přepínači

```
% \emph je příkaz s argumentem, \em je přepínač.  
\emph{zvýrazněný text}, zde je text normální % správně  
{\em emphasized text}, zde je text normální % správně  
  
\em zvýrazněný text, zde je text normální % špatně!  
\em{emphasized text}, zde je text normální % špatně!
```


Struktura zdrojového textu

```
\documentclass[volby]{trida}  
...  
... zahlavi dokumentu  
...  
\begin{document}  
...  
... textova cast  
...  
\end{document}
```

- `book` – pro sazbu knihy,
- `article` – článku,
- `report` – zprávy,
- `letter` – dopisu a
- `slides` – slajdů.

- 11pt – základní velikost (stupeň) písma celého dokumentu,
- twoside – sazba dvoustranného dokumentu,
- a4paper – změna rozměrů stránky na evropský standard,
- landscape – záměna výšky a šířky strany;
- volby se oddělují čárkami.

```
\documentclass[a4paper,10pt]{article}
```

Záhlaví dokumentu

- Globální nastavení pro celý dokument;
- připojování balíčků (package):
 - czech,
 - graphics,
 - times, palatino, bookman.

```
\usepackage[czech]{babel}  
\usepackage{times}
```

Rezervované znaky

Mají v \LaTeX u speciální význam:

\$ % ^ & _ { } ~ \

Zapíšeme je tímto způsobem:

`\# \ $ \% \^{} \& _ \{ \} \~{} \textbackslash{}`

Pozor: dvojitě zpětné lomítko ukončuje řádky tabulky (viz dále)

`\\`

První kompletní dokument

```
\documentclass[a4paper,10pt]{article}
\usepackage[czech]{babel}
```

```
\begin{document}
```

Text článku píšeme tak, jak jsme zvyklí.
Bez příkazů se sází hladká sazba.

Odstavce se oddělují prázdným řádkem,
systém sám zařídí správný zlom, odsazení
od kraje atd. Pokud chceme vyzkoušet všechny
české znaky, použijeme známou větu
Příliš žlut'oučký kuň úpěl d'ábelské ódy.

```
\end{document}
```

Obyčejná mezera, \LaTeX se postará o roztáhnutí či smrsknutí při zarovnání do bloku.

docela obyčejný text		docela obyčejný text
Hudební věda na FF		Hudební věda na FF

Používá se za tečkou, čárkou, odsuvníkem (apostrofovem), před počátečními a za koncovými uvozovkami, před a za pomlčkou (nikoli u rozsahů) atd.:

\,

3 kg		3\,kg
11.5.2011		11.\,5.\,2011

stejné vlastnosti jako nezlomitelná mezera, ale má pevné rozměry.

Nezlomitelná mezera

Se používá za jednopísmennými předložkami a spojkami: ~ (znak tilda) a zúžená nezlomitelná mezera: \, :

s dětmi o hudbě		s~dětmi o~hudbě
14 000 Kč		14\,000\,Kč
J. S. Bach		J.\,S.\,Bach
15 l rumu		15\,l rumu

Šířka mezery pomocí tildy se může měnit stejně jako mezislovní mezera, druhá varianta je pevná.

Spojovník, pomlčka (krátká, dlouhá)

Použití těchto typografických prvků známe.

bude-li	bude-li
v letech 1645–57	v~letech 1645--57
fermáta – koruna	fermáta\,--\,koruna
Já tě snad —	Já tě snad ---
–5, $a - b$	$\$-5\$$, $\$a-b\$$

Samotný znak % ve zdrojovém souboru uvozuje komentáře – nejsou vidět a \LaTeX je nezpracovává. Pro znak procenta použijeme zápis \%.

```
Johann Sebastian Bach se narodil  
31.\,3.\,1685 % ověřit  
v německém Eisenachu. 55\,\% jeho dětí  
umřelo dříve než on sám.
```

Johann Sebastian Bach se narodil 31. 3. 1685 v německém Eisenachu. 55 % jeho dětí umřelo dříve než on sám.

Používáme obyčejné a hranaté závorky zapíšeme přímo znaky na klávesnici. Pro sazbu složených závorek, které se používají pro uzavírání skupin použijeme zápis `\{` a `\}`. Závorky vypadají takto: (,), [,], { a }

Nutné používat balík `\usepackage[czech]{babel}`.

„české“		<code>\clqq české\crqq</code>
„české jednoduché“		<code>\clq české jednoduché\crq</code>
»francouzské«		<code>\flqq francouzské\frqq</code>

- ampersand & má v \LaTeX u speciální funkci, podobně jako u procenta musíme použít zápis $\backslash\&$
- paragraf § se odděluje zúženými mezerami a jeho zápis je $\backslash S$
- tři tečky (výpustek) se v některých případech odděluje zúženou mezerou, ... a tato sekvence má zápis mezerou, $\backslash, \backslash\text{dots}$
- \LaTeX ignoruje více mezer (stejně jako více prázdných řádků), za zápisem $\backslash\text{prikaz}$ musí být mezera, ale my ji v textu někdy nechceme
- znak dolaru má také speciální funkci (viz ukázka matematického zápisu), zapisujeme tedy takto: $\backslash\text{\$}$ se vysází jako $\text{\$}$

Velmi stručně o matematickém módu

TeX vytvořen primárně pro sazbu matematiky. Nejčastěji používáme tzv. odstavcový matematický mód, který se uzavírá mezi znaky $\$ \dots \$$

osobně piju jen 10° pivo	osobně piju jen 10° pivo
horní index ¹	horní index ¹
dolní index ₁₁	dolní index ₁₁
chybný index ₁₁	chybný index ₁₁

o \LaTeX u	o $\LaTeX\{ \}$ u
o \LaTeX u	o \LaTeX u
chyba!	o \LaTeX u
o \LaTeX u	o $\LaTeX\{ \}$ u
o \LaTeX u	o $\LaTeX\backslash$ u

Chyby a jejich odhalování

- špatné jméno příkazu
- chybějící ukončení skupiny

Jednoduché příkazy

ó	<code>\' {o}</code>	ò	<code>\' {o}</code>
ô	<code>\^ {o}</code>	ö	<code>\" {o}</code>
õ	<code>\= {o}</code>	ō	<code>\= {o}</code>
ò	<code>\. {o}</code>	ǒ	<code>\u {o}</code>
ǒ	<code>\v {o}</code>	ö	<code>\H {o}</code>
ô	<code>\t {oo}</code>	ô	<code>\r {o}</code>
ø	<code>\c {o}</code>	ø	<code>\d {o}</code>
ø	<code>\b {o}</code>	œ	<code>\oe</code>
æ	<code>\ae</code>	ø	<code>\o</code>
ß	<code>\ss</code>	†	<code>\dag</code>
§	<code>\S</code>	¶	<code>\P</code>
¿	<code>?'</code>	¡	<code>!'</code>

`\today`, `\copyright`, `\pounds`, `\TeX`, `\LaTeX`

18. října 2018, ©, £, T_EX, L^AT_EX

Mezery ve zdrojovém souboru

- konec řádku → mezera
- tabulátor → mezera
- sekvence bílých znaků → jedna mezera
- ignoruje mezery na konci a na začátku řádku
- všechny mezery za `\prikaz` se ignorují
- pokud chceme vysázet za příkazem mezeru, pak pomocí sekvence `_` nebo prázdnou skupinou `\prikaz{} slovo`

Příklad na použití mezer

zdroj:

```
Můžeme psát           mezery
jak                    se nám      zachce.
```

```
Dokud   nenapíšeme
                                prázdný
řádek, LaTeX sází stále týž odstavec.
```

výsledek:

Můžeme psát mezery jak se nám zachce.
Dokud nenapíšeme prázdný řádek, LaTeX sází stále týž odstavec.

Typografické jednotky v L^AT_EXu

pt	angl. typografický bod	mm	milimetr
pc	pica	dd	Didôtův bod
bp	big point (1/72")	cc	cicero
in	palec (25,4 mm)	sp	scaled point (1 pt = 65 536 sp)
cm	centimetr		
em	čtverčikový rozměr	ex	půlčtverčik

- horizontální mezery:
 - `\hspace{3cm}` – vloží vodorovnou mezeru
 - `\hspace*{15pt}` – vloží mezeru, která se zobrazí i při zlomu dlouhého řádku
- vertikální mezery:
 - `\vspace{3mm}` – vloží svislou mezeru mezi odstavce
 - `\vspace*{1em}` – vloží mezeru i na konci / začátku stránky
 - `\smallskip` ($\frac{1}{4}$ řádkování), `\medskip` ($\frac{1}{2}$) a `\bigskip` (1 řádek)

Pozor!

- každý rozměr musí mít jednotku, i nulový: `\vspace{0}` je špatně
- svislé mezery musí být mezi odstavci (před nebo za musí být prázdný řádek)

Vyplňující mezery

Pokud chcete na řádku odsunout text co nejdál doprava, případně na stránce co nejdál dolů, můžete použít příkazy `hfill` a `vfill`.

Na řádku to dopadne takto.

Na stránce zase takto.

Explicitní dělení slov

L^AT_EX dělí slova automaticky pro nastavený jazyk. Pokrytí pravidel však není 100% a proto mu někdy musíte poradit. Uděláte to příkazem `hyphenation`.

```
\hyphenation{nejku-la-t'ou-lin-ka-tější}
```

Příkaz se nejčastěji uvádí v preambuli. Můžete použít i povolení rozdělení slova příkazem `\-`:

Tak a tady mám nějaké husto\ -dé\ -monské slovo

Příkaz `\hyphenation` v kombinaci se slovy s diakritikou vyžaduje použití balíčku `fontenc`: `\usepackage[T1]{fontenc}`

Explicitní zlom řádku a stránky

- \LaTeX implicitně láme řádky podle druhu zarovnání, šířky stránky a odstavce
- pokud chceme řádek zlomit dříve / na konkrétním místě, ale nechceme začít nový odstavec, použijeme příkaz
 - `\` nebo
 - `\[rozměr]` – za zlomem vloží svislou mezeru daného rozměru
- \LaTeX láme automaticky i stránky,
- pokud chceme zlomit stránku dřív, použijeme příkaz
 - `\newpage` – veškerý obsah bude pokračovat na další stránce
 - `\clearpage` – vysází navíc všechny tabulky a obrázky, které se zatím nikam nevešly (viz dále)
 - `\cleardoublepage` – při dvoustránkové sazbě

Nastavení stupně písma

ABCDefgh	<code>\tiny</code>	ABCDefgh	<code>\large</code>
ABCDefgh	<code>\scriptsize</code>	ABCDefgh	<code>\Large</code>
ABCDefgh	<code>\footnotesize</code>	ABCDefgh	<code>\LARGE</code>
ABCDefgh	<code>\small</code>	ABCDefgh	<code>\huge</code>
ABCDefgh	<code>\normalsize</code>	ABCDefgh	<code>\Huge</code>

- Příkazy mající platnost do konce skupiny!
- každé dva různé stupně by se měly lišit alespoň o 20 %
- nepoužívat libovolné velikosti

Píšeme text a něco je `{\large větší}`.

Píšeme text a něco je **větší**.

Vyznačování

ABCDefgh	antikva (Roman)	<code>\rm</code>
<i>ABCDefgh</i>	kurzíva (Italics)	<code>\it</code>
ABCDefgh	polotučné (BOLDface)	<code>\bf</code>
<i>ABCDefgh</i>	skloněné (Slanted)	<code>\sl</code>
ABCDefgh	grotesk (SansSerif)	<code>\sf</code>
ABCDEF ^{GH}	kapitálky (Small Caps)	<code>\sc</code>
ABCDefgh	strojopisné (Typewriter)	<code>\tt</code>
<i>ABCDefgh</i>	vyznačovací (Emphasized)	<code>\em</code>

`{\it Vyznačené}` slovo se `{\bf musí}` oddělit tzv. `{\sf skupinou}`, jinak bude platit změna řezu až do konce aktuální skupiny.

Vyznačené slovo se **musí** oddělit tzv. skupinou, jinak bude platit změna řezu až do konce aktuální skupiny.

Vyznačování II

Pro změnu řezu se doporučuje používat blokové příkazy

<i>ABcd</i>		<code>\textit{ABcd}</code>
ABcd		<code>\textbf{ABcd}</code>
ABCD		<code>\textsc{ABcd}</code>

Předchozí příkazy nelze kombinovat, neboť slouží jako přepínače. Pokud chceme kombinovat sílu tahu, řez, tvar atd.

<i>ABcd</i>		<code>{\itshape ABcd}</code>
ABcd		<code>{\bfseries ABcd}</code>
ABCD		<code>{\scshape ABcd}</code>
font není dostupný		<code>{\itshape\bfseries ABcd}</code>
font není dostupný		<code>{\scshape\bfseries ABcd}</code>

Někteří mohou preferovat minuskové číslice oproti normálním. Rozdíl je patrný na první pohled a záleží na vkusu. V některých fontech však nemusí být dostupné.

1234567890 vs. 1234567890

Zápis v \LaTeX u:

1234567890 vs. `\oldstylenums{1234567890}`

Změna rozměrů stránky

Použijeme balíček `\usepackage[parameter]{geometry}`

Parametry oddělené čárkami:

- `left=2cm`
- `top=1cm`
- `right=3cm`
- `margin=2cm`
- `twoside`

Zjistěte si rozměry stránky, okraje stránky podle vnitřních předpisů FF a nastavte si hlavičku dokumentu na tyto rozměry.

Implicitně je nastaveno řádkování 1, vhodné v rozsahu 1–1,5. Změní se pomocí příkazů v záhlaví:

- definice rozměru `\baselineskip`:
`\baselineskip=15pt`
nastaví řádkování 15 pt, což znamená při stupni písma 10 pt řádkování 1,5
- redefinice natáhnutí meziřádkové mezery:
`\def\baselinestretch{1.5}\normalsize`
nastaví řádkování 1,5 podle základního stupně písma
- příkazu `\linespread{1.3}`

Rozměry a vlastnosti odstavce lze upravovat příkazy v záhlaví dokumentu, případně uvnitř skupiny:

- `\setlength{\parindent}{1em}` – odstavcová zarážka 1 em
- `\setlength{\parskip}{5pt}` – vzdálenost mezi odstavci
- `\setlength{\leftskip}{1ex}` – šířka levého okraje
- `\setlength{\rightskip}{10mm}` – pravý okraj
- `\noindent` – odstraní odstavcovou zarážku
- `\indent` – vytvoří zarážku

- Ohraničeny buď složenými závorkami `{, }` nebo
- dvojicí příkazů `\begin{název}` a `\end{název}`.
- Druhý způsob představuje tzv. prostředí (obrázek, tabulka, dokument, seznam, ...),
- některé příkazy ovlivňují sazbu až do konce skupiny (prostředí), v němž jsou uvedeny, hranicemi skupin tedy nastavujeme rozsah vlivu tohoto příkazu;
- skupiny se mohou libovolně vnořovat, nesmějí se však křížit.

Někdy je vhodnější použít předdefinované prostředí, které mění vlastnosti odstavce.

- prostředí jsou skupiny a mají zápis

```
\begin{prostředí}
```

```
...
```

```
\end{prostředí}
```

- `quotation` – nastaví okraje odstavce (odsadí)
- `quote` – jako `quotation`, ale bez zářezek
- `flushleft` – zarovnání na levý praporek
- `flushright` – zarovnání na pravý praporek
- `center` – zarovnání na střed
- `verbatim` – \LaTeX sází přesně to, co je ve zdrojovém souboru

Seznamy, číslované seznamy, seznamy definic, vnořené seznamy.

Jednotlivé položky se oddělují příkazem `\item`. Za ním může následovat nepovinný parametr `\item [par]`, kde `par` odpovídá značce před položkou.

Jde o prostředí, takže se uzavírají do dvojice příkazů

`\begin{env}` a `\end{env}`

kde `env` odpovídá `itemize`, `enumerate` a `description`.

Nečíslovaný seznam

```
\begin{itemize}
  \item První položka,
  \item[$\cdot$] druhá položka a
  \item[---] třetí položka.
\end{itemize}
```

- První položka,
- druhá položka a
- třetí položka.

```
\begin{enumerate}  
  \item První položka,  
  \item druhá položka a  
  \item třetí položka.  
\end{enumerate}
```

- 1 První položka,
- 2 druhá položka a
- 3 třetí položka.

Seznam pojmů

```
\begin{description}
  \item[První] První pojem.
  \item[{\bf Druhý}] \ \ \
    Druhý pojem.
  \item[Třetí] \ \ \
    Třetí pojem.
\end{description}
```

První První pojem.

Druhý

Druhý pojem.

Třetí

Třetí pojem.

```
\begin{enumerate}
  \item První položka,
  \begin{itemize}
    \item První vnořená nečíslovaná,
    \item Druhá vnořená nečíslovaná.
  \end{itemize}
  \item Druhá položka a
  \item Třetí číslovaná položka.
\end{enumerate}
```

- 1 První položka,
 - První vnořená nečíslovaná,
 - Druhá vnořená nečíslovaná.
- 2 Druhá položka a
- 3 Třetí číslovaná položka.

Pomocí balíčku `color` a `xcolor`, tedy

```
\usepackage{color} nebo
```

```
\usepackage[usenames,dvipsnames,svgnames,table]{xcolor}
```

Obarvení **slova** ve větě se pak provede příkazem

```
\textcolor{red}{slova}
```

Můžeme používat předdefinované barvy `white`, `black`, `red`, `green`, `blue`, `cyan`, `magenta`, `yellow`, pokud použijeme v deklaraci balíčku `xcolor` parametr `dvipsnames`, budeme mít k dispozici další barvy.

Pomocí příkazu `\definecolor` si můžeme definovat barvu vlastní:

```
\definecolor{mojebarvicka}{RGB}{155,201,25} nebo
```

```
\definecolor{mojebarvicka}{rgb}{0.4,0.1,0.9}
```

```
\textcolor{mojebarvicka}{Tohle je moje barvička!}
```

Tohle je moje barvička!

Definovány v hlavičce dokumentu:

```
\documentclass[a4paper,12pt]{article}
```

article – články, nejčastěji používaná třída,

report – varianta předchozího,

letter – dopisy,

book – kniha s kapitolami,

proc – sborníky z konferencí, složené z článků a

slides – slajdy, lépe použít **beamer** – o tom později.

Máme k dispozici několik příkazů s obecným tvarem `\titulek[text v obsahu a záhlaví]{text}`. Tyto příkazy vysází nadpis oddílu a postarají se o veškeré formátování.

- `\part`
- `\chapter`
- `\section`
- `\subsection`
- `\subsubsection`

Varianta s hvězdičkou `\titulek*`, který nepřidá nadpis (oddíl) do obsahu a nesází číslo oddílu.

```
\chapter[0 prasátku]{Kterak se prasátko vydalo do světa}
```

Titulek

```
\documentclass[11pt,a4paper]{report}

\title{0 \LaTeX{}u}
\author{Vít Baisa\\
Masarykova univerzita, Brno\\
{\tt vit.baisa@gmail.com}}

\begin{document}
\maketitle

...
\end{document}
```

V místě, kde chceme poznámku¹ uvedeme příkaz `\footnote{Doplňující text.}`. \LaTeX sám použije nejnižší dosud nepoužité číslo poznámky, vysází index a doplňující text vysází na konci odpovídající stránky.

V některých prostředích (tabular, viz dále) poznámky nefungují, je potřeba použít alternativní příkazy. Vždy však lze konkrétní příklady obejít vhodnou formulací v textu.

¹Ke které je i doplňující text.

Více sloupců

Makra v balíčku `\usepackage{multicol}`. Prostředí `multicols` s parametrem pro počet sloupců – ty jsou vyvážené.

```
\begin{multicols}{3}
```

Lorem ipsum dolor sit
amet, consectetur
adipiscing elit.

Suspendisse
scelerisque, sapien eu
iaculis varius, sem
ligula venenatis lorem,
sit amet rhoncus
magna urna eu libero.

```
\end{multicols}
```

Proin aliquet aliquam
ultrices. Nam id leo vel
metus porttitor cursus.

In hac habitasse platea
dictumst. Donec
viverra malesuada
semper. Quisque eu
commodo ante. Nam
fermentum mi at dolor

semper auctor cursus
mi faucibus. Aliquam
erat volutpat. Aenean
suscipit, odio ac iaculis
placemat, ligula libero
adipiscing nisi,
accumsan consequat
lacus purus id sapien.

K sazbě tabulek slouží prostředí `tabular`. \LaTeX určuje šířku sloupečků automaticky. Obecný tvar začátku prostředí je

```
\begin{tabular}{specifikace_sloupcu}
```

Specifikace může obsahovat tyto prvky:

<code>l</code>	zarovnání sloupce doleva
<code>c</code>	dtto na střed
<code>r</code>	dtto doprava
<code>p{šířka}</code>	odstavcový sloupec
<code> </code>	svislá čára
<code> </code>	dvojitá svislá čára

Tabulky – příkazy

Uvnitř prostředí (tabulky) pak vkládáte samotný text. Sloupce oddělujete znakem `&`.

<code>\hline</code>	Vysází horizontální čáru,
<code>\\</code>	ukončí řádek,
<code>\newline</code>	zlomí řádek,
<code>\cline{n-m}</code>	vykreslí horizontální čáru od n. sloupce po m. sloupec

Pozor na to, že `\cline` v kombinaci s `babel` balíčkem a nastavením `czech` způsobuje problémy. Je potřeba změnit typ znaku – takto:

```
\usepackage{etoolbox}
\preto\tabular{\shorthandoff{-}}
```

Tabulky

```
\begin{tabular}{|r|l|}
položka & a druhá
další & {\bf tučně}
\multicolumn{2}{|c|}{\it na střed}
sloučené &
řádky &
\end{tabular}
```

položka	a druhá
další	tučně
<i>na střed</i>	
sloučené řádky	

Sloučené sloupce a řádky

V tabulkách můžeme slučovat sloupce pomocí příkazu `\multicolumn{pocet}{format}{obsah}` či slučovat řádky za použití balíčku `\usepackage{multirow}` a příkazem

`\multirow{pocet}{šířka}{obsah}`

A	B	C	D
EF		G	H
IM	J	K	L
	N	O	P

```
\begin{tabular}{|c|c|c|c|}\hline
A & B & C & D \\ \hline
\multicolumn{2}{|c|}{EF} & G & H \\ \hline
\multirow{2}{*}{IM} & J & K & L \\
& N & O & P \\ \hline
\end{tabular}
```

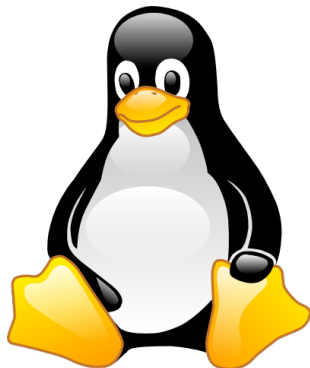

- balíček `\usepackage{graphicx}`
- příkaz `\includegraphics[volby]{nazev_obrazku}`
- volby jsou nepovinné, jednotlivé dvojice klíč=hodnota se oddělují čárkou

klíč	zápis	význam
angle	angle=25	rotace kolem středu
height	height=15mm	výška obrázku
width	width=12cm	šířka obrázku
scale	scale=0.5	změna velikosti násobkem

Podporované formáty (pdf_latex, pdf_cslatex): JPEG, PNG, PDF.

Ukázka vložení obrázku

```
\includegraphics [width=4cm] {tucnak.png}
```



Plovoucí prostředí

Blok, který může plavat a hledat si nejvhodnější místo v sazbě tak, aby nevznikaly velké mezery. Zároveň nesmí být rozlomen na více částí (obrázek, tabulka).

K tomu slouží dvě prostředí: `figure` a `table`. Jimi ohraničujeme tabulky (`tabular`) nebo obrázky (`includegraphics`).

Jejich umístění a vzhled můžeme ovládat příkazy a volitelnými parametry:

- 1–4 znaky pro umístění `\begin{figure}` [umístění]:
 - `h` (here) – objekt se umístí přednostně do místa, kde je ve zdrojovém souboru
 - `t` (top) – do horní části stránky
 - `b` (bottom) – do dolní části stránky
 - `p` (page) – na stranu, která obsahuje pouze plovoucí prostředí
 - příklad: `\begin{table}[ht]...`
- `\center` – na začátku skupiny, vysází objekt na střed
- `\caption{popis}` – nad nebo pod obrázkem / tabulkou: vysází text popisu s číslem obrázku nad / pod objektem.

Tabulka v plovoucím prostředí

```
\begin{table}[specifikátor pozice]
\centering
\begin{tabular}{|1|}
... tabulka ...
\end{tabular}
\caption{Tato tabulka obsahuje nějaká data}
\label{mojetabulka}
\end{table}
```

Křížové odkazy

Odkazujeme číslem strany (*viz str. X*) nebo uvedením čísla objektu (*jak jsme uvedli v kapitole 4.1*)

Příkazem `\label{odkaz}` označíme místo, na které chceme odkazovat.

\LaTeX si zapamatuje: v prostředí číslování, mimo prostředí číslo sekce.

- `\ref{odkaz}` vysází číslování (číslo (podkapitoly), číslo v číslovaném seznamu, číslo obrázku apod.)
- `\pageref{odkaz}` vysází stránku, na které se daný objekt vyskytuje

Jako odkaz slouží jednoznačný identifikátor (bez diakritiky).

Typické použití:

```
\chapter{Na sklonku života} \label{skl}
```

```
Když už věděl, že brzy zemře...
```

```
...
```

```
Kapitola~\ref{skl} na str.\, \pageref{skl} se věnuje...
```

Obsah – seznam kapitol, podkapitol atd.

\LaTeX vytváří seznam literatury automaticky – podle použití titulků a oddílů. Samotný obsah se vysází příkazem `\tableofcontents`. Ten může být kdekoli ve skupině `document`. Pokud je obsah na začátku, musí se zdrojový soubor přeložit až 3×.

Tento příkaz funguje pouze ve třídě `book` a `report`.

- první překlad: \LaTeX si ukládá seznam oddílů (soubor `.toc`)
- druhý překlad: \LaTeX vysází tento seznam do těla dokumentu
- třetí překlad: někdy je nutné přepočítat stránky, protože obsah na začátku změnil stránkování

Další praktické příkazy:

- `\listoffigures` – vysází seznam všech obrázků (`figure`)
- `\listoftables` – seznam všech tabulek (`table`)
- `\addcontentsline{toc}{titulek}{text}`
vloží do obsahu `text` a stránku pro nečíslovaný titulek (`\titulek*`)

Pro kompilaci rejstříku budeme potřebovat minimálně 4 příkazy.

- 1 načtete nový balík `makeidx`,
- 2 v hlavičce použijte příkaz `\makeindex` pro inicializaci; při překladu se pak vytvoří pomocný soubor s příponou `idx`,
- 3 v místě, kde chcete rejstřík umístit použijte příkaz `\printindex`
- 4 a příkaz pro přidání pojmu do rejstříku: `\index{pojem}`

Pojmy musíme zpracovat příkazem `makeindex` soubor. Tento příkaz zpracuje soubor `soubor.idx`, který je vytvořen při spuštění `pdfcslatex`. Poté znovu spusťte `pdfcslatex`. Rejstřík by měl nyní být vysázen v místě použití příkazu `\printindex`.

```
pdfcslatex soubor
makeindex soubor
pdfcslatex soubor
pdfcslatex soubor
```

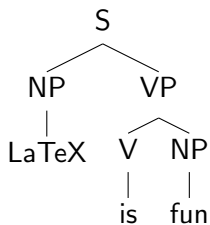

Formátování rejstříku

Rejstřík můžeme ovládat pomocí následujících variant příkazů.

<code>\index{jméno}</code>	jméno, 1	jednoduchá položka
<code>\index{jméno!Pavel}</code>	Pavel, 3	podpojem
<code>\index{Honza@\textsl{Honza}}</code>	<i>Honza</i> , 2	formátování
<code>\index{Petr@\textbf{Petr}}</code>	Petr , 7	dtto
<code>\index{Jarek textbf}</code>	Jarek, 3	formát čísla stránky
<code>\index{Zuzka textit}</code>	Zuzka, 5	dtto
<code>\index{Karel see{Jarek}}</code>	Karel, viz Jarek	křížové odkazy
<code>\index{A seealso{B}}</code>	A, viz také B	dtto
<code>\index{A ()... \index{A)}</code>	A, 19–32	vícestránkový pojem

Více rejstříků

```
\usepackage{multind}  
\makeindex{knihy}  
\makeindex{autori}  
...  
\index{knihy}{Položka do rejstříku knih}  
\index{autori}{Do druhého indexu}  
...  
\printindex{knihy}{Rejstřík knih}  
\printindex{autori}{Rejstřík autorů}
```



Obrázek: Složkový stromeček

Seznam literatury a citace

Existuje více způsobů, zde nejjednodušší. Citace podle české normy. Na konci souboru, kde chcete mít seznam literatury použijeme prostředí `thebibliography`:

```
\begin{thebibliography}{nejširší_položka}
\addcontentsline{toc}{chapter}{Seznam literatury}
\bibitem[text]{odkaz}
Zde je text citace, formátován podle požadované normy.
\bibitem{odkaz-druhy} ...
\end{thebibliography}
```

V textu kde chceme odkázat na literaturu použijeme příkaz `\cite{odkaz}`. Funguje podobně jako příkaz `\label`, ale jako text zobrazuje specifikaci uvedenou v seznamu literatury.

```
\renewcommand{\refname}{Použitá literatura}
```

Balíček natbib

Implicitní formát citací je číslování a uzavření do hranatých závorek. Příkaz `\cite{xxx}` vysází např. [23].

Text citace můžete změnit volitelným parametrem `\cite[str.~45]{xxx}` vysází [23, str. 45].

Další úpravy umožní použití a nastavení balíčku natbib spolu s novými příkazy.

```
\usepackage[options]{natbib}
```

Citation command	Output
<code>\citet{goossens93}</code>	Goossens et al. (1993)
<code>\citep{goossens93}</code>	(Goossens et al., 1993)
<code>\citet*{goossens93}</code>	Goossens, Mittlebach, and Samarin (1993)
<code>\citep*{goossens93}</code>	(Goossens, Mittlebach, and Samarin, 1993)
<code>\citeauthor{goossens93}</code>	Goossens et al.
<code>\citeauthor*{goossens93}</code>	Goossens, Mittlebach, and Samarin
<code>\citeyear{goossens93}</code>	1993
<code>\citeyearpar{goossens93}</code>	(1993)
<code>\citealt{goossens93}</code>	Goossens et al. 1993
<code>\citealp{goossens93}</code>	Goossens et al., 1993

Citace je někdy lepší mít uložené v externím souboru s příponou `.bib`.
Jedna položka může vypadat zhruba takto:

```
@article{greenwade93,  
  author   = "George D. Greenwade",  
  title    = "The Comprehensive Tex Archive Network",  
  year     = "1993",  
  journal  = "TUGBoat",  
  volume   = "14",  
  number   = "3",  
  pages    = "342--351"  
}
```

Poté stačí uvést na konci dokumentu příkazy

```
\bibliographystyle{plain}  
\bibliography{citace}
```

Pro použití jednoduchého formátování a citací uložených v souboru `citace.bib`.

Cituje se jak jsme zvyklí, pomocí příkazu `\cite{ref}`.

Je potřeba spustit tuto sekvenci příkazů:

```
pdfcslatex soubor.tex  
bibtex soubor.tex  
pdfcslatex soubor.tex  
pdfcslatex soubor.tex
```

Soubory pro L^AT_EX

- .aux pomocný soubor obsahující křížové odkazy
- .bbl výstup BibTexu pro použití L^AT_EXem
- .bib soubor se seznamem literatury ve speciálním formátu
- .bst soubor se stylem pro BiBTeX
- .cls třída pro použití s příkazem `documentclass`
- .dvi Device Independent soubor, hlavní výstup L^AT_EXu
- .pdf Portable Document Format, zobrazitelný výstup
- .log soubor obsahující detaily o průběhu posledního překladu
- .toc soubor obsahující nadpisy kapitol
- .lof seznam obrázků
- .lot seznam tabulek
- .idx seznam položek z rejstříku, lze zpracovat příkazem `makeindex`
- .ind zpracovaný soubor .idx, použitelný pro L^AT_EX
- .sty makro balíček, možno načíst příkazem `usepackage`
- .tex zdrojový soubor pro L^AT_EX nebo T_EX