

# Scucnutý úvod do L<sup>A</sup>T<sub>E</sub>Xu pre programátorov

Kubo

4. decembra 2005

## 1 Úvod

T<sub>E</sub>X<sup>1</sup> je typografický systém programátorského guru menom Don Knuth. Ten už predtým trochu fušoval do typografie a keď videl ako sa kvalita (typografická, samozrejme) jeho článkov a obzvlášť jeho kníh (*The Art of Computer Programming*) zhoršuje, povedal si „tak dosť“ a spravil si na to vlastný program. Po piatich rokoch v 1982-om dokončil T<sub>E</sub>X, ktorý s drobnými úpravami v 1989 (napr. prechod na 8 bitov) používame dodnes! Vskutku nie je veľa programov podobného formátu. Za to, že je ešte dnes nielenže použiteľný, ale vysoko kvalitný nástroj, vďačí aj tomu, že je programovateľný (hoci to nebol zámer, T<sub>E</sub>X je ekvivalentný Turingovmu stroju). T<sub>E</sub>X nie je WYSIWYG, ale makroprocesor, ktorý načítava vstupný text (spolu s príkazmi, ako text upraviť), expanduje makrá a primitívne príkazy vykonáva.

L<sup>A</sup>T<sub>E</sub>X je balík makier pôvodne od Leslieho Lamporta, kde sú definované high-levelové makrá popisujúce vzhľad dokumentu. Hneď na začiatku treba upozorniť, že L<sup>A</sup>T<sub>E</sub>X nie je úplne kompatibilný s pôvodným T<sub>E</sub>Xom (presnejšie s tzv. plain T<sub>E</sub>Xom; plain je štandardný balík makier priamo od Knutha). Výhodou L<sup>A</sup>T<sub>E</sub>Xu je veľká pohodlnosť – ľahko sa vygenerujú aj zložitejšie prvky ako obsah, register, zoznam použitej literatúry a rôzne odkazy; tiež sú v ňom preddefinované rôzne štýly, ktoré sa starajú o vzhľad stránok textu. Na druhej strane je treba povedať, že na rozdiel od samotného T<sub>E</sub>Xu L<sup>A</sup>T<sub>E</sub>X nie je stabilný, ešte stále sa vyvíja a niekedy dosť ťažkopádny (preto je dobre naučiť sa aj T<sub>E</sub>X samotný).

## 2 Čučoriedkový koláč

Je čas ukázať si prvý dokument. Bude vyzeráť asi takto:

```
\documentclass{article}
\usepackage{slovak}
\begin{document}
Čučoriedkový koláč.
\end{document}
```

Tento text budeme písať do obyčajného textového editora (bez formátovania! žiadny Word!), ideálne s highlightovaním syntaxu (napr. vim). Po uložení text prebehne programom `cslatex`, ktorý ho „zošrotí“ a vytvorí device independent súbor (`.dvi`; dá sa prezerať programami ako `xdvi`, `yap`, `windvi`) – ten môžeme nástrojom `dvips` skonvertovať do postscriptu (`.ps`; dá sa prezerať pomocou `ghostscriptu`). Iná možnosť je programom `pdfcslatex` vytvoriť priamo pdf-dokument.

Samotný dokument začína príkazom `\begin{document}` a končí `\end{document}`. Text medzi tým sa naozaj zobrazí vo výstupnom súbore (v štandardnom fonte, ktorým je Computer Modern Roman, rovnako z dielne uja Knutha a spol.; navyše je odsadený doprava ako začiatok odstavca). Časť predtým voláme preambula a píšeme tam rôzne nastavenia, definujeme nové príkazy a podobne. Konkrétne príkaz `\documentclass` nastavuje štýl dokumentu (v tomto prípade je to `article`, teda článok), `\usepackage{slovak}` includeje slovenčinový balík.

---

<sup>1</sup>čítaj „tech“

### 3 Príkazy a prostredia

S príkazmi a prostrediami sme sa stretli už v prvom príklade. Tu si prezradíme, čo sa v  $\LaTeX$ u považuje za príkaz a čo za prostredie<sup>2</sup>.

Príkazy sú dvoch druhov: tzv. control symbols a control sequences. Všetky príkazy začínajú backslashom ( $\backslash$ ). Ak za ním nasleduje písmeno, za príkaz sa považuje postupnosť písmen (po najbližšie nepísmeno; case sensitive). Ak za backslashom nasleduje iný symbol, za príkaz sa považuje tento jediný symbol. Napríklad v  $\TeX$  je `sch\on\ldots`, čo vysádza „ $\TeX$  je schön...“, sa nachádzajú dve control sequence: `\TeX` (postupnosť písmen, končí ďalším backslashom) a `\dots` a dva control symboly: `\_` a `\`, ktoré sa starajú o medzeru za slovom  $\TeX$  a o prehlasovanie o-čka.

Príkazy môžu mať aj parametre. V  $\LaTeX$ u existujú povinné aj nepovinné parametre. Povinné sa väčšinou píšú do skladaných zátvoriek (`{...}`), nepovinné do hranatých zátvoriek (`[...]`). Vezmime si napríklad hneď úvodný príkaz `\documentclass` – ten nastavuje celkový vzhľad (štýl) dokumentu. Na výber máme štandardne napr. `article`, `report`, či `book`. Do hranatých zátvoriek môžeme tiež uviesť nepovinné parametre oddelené čiarkou, napr.

```
\documentclass[11pt,a4paper]{article} je článok 11-bodovým písmom na a4
\documentclass[12pt,twocolumn]{book} kniha 12-bodovým písmom; text je v dvoch stĺpcoch
\documentclass[twoside]{report} je obojstranný report
```

Špeciálne príkazy sú `\begin` a `\end`. Tie majú ako parameter názov prostredia, ktoré ohraničujú. Už sme videli prostredie `document`, ktoré musí každý  $\LaTeX$ ovský dokument obsahovať. Prostredia fungujú podobne ako párové tagy v HTML – môžeme ich do seba vnárať, ale nesmieme ich krížiť.

### 4 Riadky a medzery

$\TeX$  má za úlohu vysádzať text čo najkrajšie; preto sú riadky a medzery na výstupe iné ako v pôvodnom dokumente.  $\TeX$  považuje koniec riadku ako medzeru – miesta, kde zalomiť riadok si  $\TeX$  hľadá sám (dokonca rozdelí slovo, aby riadky neboli príliš riedke alebo husté). Ak medzi nejakými dvoma slovami nechceme zalomiť riadok, použijeme namiesto obvyčajnej medzery nedeliteľnú (`~`). Napríklad ak nechceme, aby z `obr. 2` bol `obr.` na konci riadku a `2` na novom, píšeme `obr.~2`. Naopak, ak chceme niekde riadok zalomiť, stačí napísať `\\` alebo `\newline` (podobne `\newpage` prejde na novú stranu).

Medzery na začiatku riadku úplne  $\TeX$  ignoruje – ak chceme nový paragraf, stačí vynechať riadok (paragraf sa automaticky odstaď doprava). Viacero voľných riadkov, či viacero medzier sa má rovnaký účinok ako jediná. Za príkazmi  $\TeX$  všetky medzery ignoruje – „v  $\TeX$ u“ teda môžeme napísať ako v  $\TeX$  u (pozor, nie v  $\TeX$ u!  $\TeX$  by považoval  $\TeX$ u za príkaz). Ak chceme predsalen za slovom  $\TeX$  medzeru (alebo viac medzier za sebou), môžeme použiť napríklad „tvrdú medzeru“ – príkaz `\_` ako v  $\TeX\_je$ .

### 5 Špeciálne znaky

V  $\TeX$ u sa niektoré znaky používajú pre špeciálne účely (z nich už poznáme `\`, `~` { a } ; hranaté zátvorky nie sú špeciálne!). Ostatné špeciálne znaky sú `#`, `$`, `%`, `^`, `&` a `_`. Ich význam si vysvetlíme neskôr. Ak by sme však tieto znaky chceli napísať, tak mriežku vypíšeme pomocou `\#`, dolár pomocou `\$`, striešku pomocou `\~{}` alebo `\^{\_}` (strieška sa v texte používa na vysádzanie cirkumflexu: napr. `\~e` je francúzske `ê`; iný význam si ukážeme neskôr), ampersand pomocou `\&`, podčiarovník pomocou `\_`, zložené zátvorky pomocou `\{` a `\}` a tildu pomocou `\~{}` alebo `\^{\_}` (`\~n` je španielske `ñ`). S backslashom je to zložitejšie, jedna možnosť je napísať `\backslash$`.

<sup>2</sup>tu som trochu váhal, ako to napísať...  $\LaTeX$  je totiž iba makro-balík,  $\TeX$  je ten, čo všetko vykonáva; na druhej strane,  $\TeX$  sa dá do veľkej miery upraviť,  $\LaTeX$  je toho dôkazom: v samotnom  $\TeX$ u neexistujú prostredia, na začiatok netreba písať `\documentclass` a.p.

V typografií sa rozlišuje medzi viacerými druhmi pomlčiek. Je treba rozlišovať minimálne spojovník (ako-tak, sem-tam; píšeme ako-tak) a pomlčku (strany 17–22; píšeme 17--22). V  $\text{T}_{\text{E}}\text{X}$ u existuje ešte (---) dlhšia pomlčka—tá sa však v slovenčine nepoužíva (môže sa však hodiť na uvádzanie priamej reči).

Ďalšie špeciálne znaky sú uvedené v tabuľke.

$\backslash\text{TeX}$	$\text{T}_{\text{E}}\text{X}$	$\backslash\text{dots}$	...	$\backslash\text{o}$	ø	$\backslash\{e\}$	é	$\backslash\{g\}$	ğ
$\backslash\text{LaTeX}$	$\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$	$\backslash\text{oe}$	œ	$\backslash\text{O}$	Ø	$\backslash\{a\}$	à	$\backslash\{v\{a\}$	ã
$\backslash\text{dag}$	†	$\backslash\text{OE}$	Œ	$\backslash\text{l}$	ł	$\backslash\{e\}$	ê	$\backslash\text{H}\{o\}$	ő
$\backslash\text{ddag}$	‡	$\backslash\text{ae}$	æ	$\backslash\text{L}$	Ł	$\backslash\{a\}$	ä	$\backslash\text{t}\{oo\}$	ôö
$\backslash\text{S}$	§	$\backslash\text{AE}$	Æ	$\backslash\text{ss}$	ß	$\backslash\{n\}$	ñ	$\backslash\text{c}\{c\}$	ç
$\backslash\text{P}$	¶	$\backslash\text{aa}$	å	$\backslash\text{i}$	ı	$\backslash\{a\}$	ā	$\backslash\text{d}\{a\}$	ą
$\backslash\text{copyright}$	©	$\backslash\text{AA}$	Å	$\backslash\text{j}$	ĵ	$\backslash\{a\}$	à	$\backslash\text{b}\{a\}$	ą

V posledných dvoch stĺpcoch sú príkazy, ktoré nad znaky sádzajú tzv. akcenty. Príkazy  $\backslash\text{i}$  a  $\backslash\text{j}$  na i a j bez bodky sa hodia, keď nad tieto písmená potrebujeme vysádzať nejaký akcent (napr. naíve sa píše  $\text{na}\backslash\text{i}$  ve, nie  $\text{na}\backslash\text{ive}$  – príkaz  $\backslash\text{ive}$  neexistuje, ani  $\text{na}\backslash\text{ive}$  – naíve). Samozrejme, keď píšeme slovenský text, nemusíme písať dlžne a mäkkéne takto ťažkopádne cez príkazy. Vďaka tomu, že používame počeskoslovenčenú verziu ( $\text{cslatex}$  alebo  $\text{pdfcslatex}$ ) a na začiatok píšeme  $\backslash\text{usepackage}\{\text{slovak}\}$ , môžeme písať jednoducho áéíóú namiesto  $\backslash\text{a}\backslash\text{e}\backslash\text{i}\backslash\text{o}\backslash\text{u}$ , čšž namiesto  $\backslash\text{v}\ \text{c}\backslash\text{v}\ \text{s}\backslash\text{v}\ \text{z}$ , äö namiesto  $\backslash\text{a}\backslash\text{o}$  (nehovoriac o  $\text{ďtř}$ , kde  $\backslash\text{v}\{\text{d}\}\backslash\text{v}\{\text{t}\}\backslash\text{v}\{\text{l}\}$  dáva  $\text{ďtř}$ ). Tiež tu je definovaný príkaz  $\backslash\text{r}$  (napr.  $\backslash\text{r}\{\text{u}\}$ ), ktorý sádza český krúžok (ů).

## 6 Užitočné príkazy...

### 6.1 Nadpis

Na začiatku som jednoducho napísal  $\backslash\text{title}\{\text{Scucnutý úvod do } \backslash\text{LaTeX} \text{ u pre programátorov}\}$ , čím som  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ u oznámil, ako sa toto tu bude volať. Za tým nasledovalo  $\backslash\text{author}\{\text{Kubo}\}$ , čím som  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ u oznámil svoje ctené meno. Nakoniec som už len napísal príkaz  $\backslash\text{maketitle}$  a  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$  za mňa nastavil písmo, vypísal nadpis, vyriešil (vertikálne) medzery a sám doplnil dátum výroby. Ak by sme chceli iný dátum, stačí nastaviť  $\backslash\text{date}\{\text{24.12.2000}\}$ . Ak nič nenapíšeme, bude nastavené  $\backslash\text{date}\{\backslash\text{today}\}$  (príkaz  $\backslash\text{today}$  sa nahradí dnešným dátumom); ak žiadny dátum nechceme, nastavíme  $\backslash\text{date}\{\}$ .

### 6.2 Členenie textu

Samotný text sa väčšinou člení (čím technickejší text, tým viac ;) ) na kapitoly, sekcie, podsekcie... V  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ u na to existujú príkazy  $\backslash\text{part}$ ,  $\backslash\text{chapter}$ ,  $\backslash\text{section}$ ,  $\backslash\text{subsection}$ ,  $\backslash\text{subsubsection}$ ,  $\backslash\text{paragraph}$  a  $\backslash\text{subparagraph}$  (v tomto texte som použil iba  $\backslash\text{section}$  a  $\backslash\text{subsection}$ ). Parameter je názov, napríklad tento odsek začal príkazom  $\backslash\text{subsection}\{\text{Členenie textu}\}$ . Príkaz  $\backslash\text{chapter}$  sa dá použiť iba v  $\text{book}$  a  $\text{report}$  – v  $\text{article}$  neexistuje. Výhodou tohoto prístupu je, že stačí napísať čarovné slovíčko  $\backslash\text{tableofcontents}$  a  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$  sám vyprodukuje obsah (pozor, dokument treba potom niekedy aspoň dvakrát prejsť  $\text{cslatex-om}$ ).

### 6.3 Fonty

Ak chcete nejaký text *zvýrazniť*, stačí jednoducho  $\text{text } \backslash\text{emph}\{\text{zvýrazniť}\}$ . Odporúčam používať výsostne tento spôsob. Príliš veľa druhov písma textu skôr uškodí, bude „krikľavý“. Zvýrazniť všetko znamená nezvýrazniť nič. Ak predsalen niektorý font potrebujete, zhrňa ich nasledujúca tabuľka.

$\backslash\text{emph}\{\text{emphasize}\}$	<i>emphasize</i>	$\backslash\text{textrm}\{\text{roman}\}$	roman
$\backslash\text{textit}\{\text{italic}\}$	<i>italic</i>	$\backslash\text{textsl}\{\text{slanted}\}$	<i>slanted</i>
$\backslash\text{textsf}\{\text{sans serif}\}$	sans serif	$\backslash\text{textbf}\{\text{bold face}\}$	<b>bold face</b>
$\backslash\text{texttt}\{\text{typewriter}\}$	typewriter	$\backslash\text{textsc}\{\text{Small Caps}\}$	SMALL CAPS

## 6.4 Iné

Veľmi užitočný príkaz je `\footnote{text}`<sup>3</sup>, ktorý *text* vysádza ako poznámku pod čiarou. Tento príkaz sa píše hneď za slovom, za ktorým má byť číslo poznámky.

Programátori určite ocenia príkaz `\verb`, ktorý vypíše všetko tak, ako je (navyše typewriterovým fontom). V tomto texte je použitý na písanie príkazov v  $\text{\LaTeX}$ u. Napr. (napr. `\r{u}`) sa dá napísať ako (napr. `\verb!\r{u}!`). Text, ktorý sa vypíše tak, ako je, končí rovnakým znakom, akým začína (v našom príklade je `\r{u}` ohraničené výkričníkmi). Ak chceme vysádzať viditeľné medzery (medzery<sub>u</sub>su<sub>u</sub>dolezite), použijeme `\verb*` (`\verb*'medzery su dolezite'` – tu je text ohraničený apostrofmi). Celý `\verb` sa musí nachádzať na jednom riadku. Pozri tiež prostredie `verbatim`.

## 6.5 Balíky

Toto, samozrejme, nebol vyčerpávajúci zoznam príkazov. Napriek tomu sa môže stať, že kýžený príkaz nie je ani vo vyčerpávajúcom zozname. Vtedy sa môžete obrátiť na nejakého  $\text{\TeX}$ ového guru, ktorý vám taký príkaz napíše, alebo sa poobzerať po internete, napr. na <http://www.ctan.org/tex-archive/macros/latex/contrib/> po vhodnom makro-balíku. Napr. v tomto texte je použitý poslovenčujúci balík `slovak` a `a4wide`, ktorý nastaví väčšiu šírku textu.

Balík sa dá použiť príkazom `\usepackage`, ako sme videli už v druhom odseku.

## 7 ... a prostredia

### 7.1 Zarovnanie

Štandardne  $\text{\TeX}$  zarovnáva na oba okraje. Na zarovnanie vľavo, vpravo alebo do stredu slúžia prostredia `flushleft`, `flushright` a `center`. Tzn., že ak chceme text zarovnaný na stred, napíšeme pred neho `\begin{center}` a na koniec `\end{center}`.

### 7.2 Abstrakt

Abstrakt sa píše na začiatok dokumentu do prostredia `abstract` (nie `abstrakt`; v poslovenčenej verzii sú príkazy rovnaké, iba sú predefinované, aby vypisovali slovenské hlášky, teda vypíše sa naozaj „abstrakt“).

### 7.3 Verbatim

Podobne ako príkaz `\verb`, resp. `\verb*`, prostredie `verbatim`, resp. `verbatim` zobrazuje text tak, ako je. Prostredie však slúži na dlhšie texty – príklad `verbatim`-u je hneď v druhom odseku – Čučoriedkový koláč.

### 7.4 Zoznamy

Prostredie `enumerate` vytvorí číslovaný zoznam, `itemize` nečíslovaný. V týchto prostrediach príkaz `\item` začne novú položku (nepovinný parameter je znak odrážky, ako ukazuje štvrtý `\item`):

```
\begin{enumerate}
  \item Prvý \item Druhý
  \begin{itemize}
    \item tretí
    \item[--] štvrtý
  \end{itemize}
\end{enumerate}
```

1. Prvý  
2. Druhý  
• tretí  
– štvrtý

---

<sup>3</sup> *text*