



## Začínáme s $\text{T}_{\text{E}}\text{X}$ em na math.muni.cz

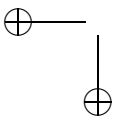
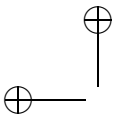
Roman Plch

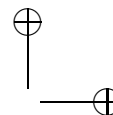
září 2004

---

*Tento dokument se snaží pomoci těm z vás, kteří mají zájem začít používat instalaci sázečního systému  $\text{T}_{\text{E}}\text{X}$  na math.muni.cz. Taktéž může pomoci i těm, kteří hledají odpověď na některý specifický, s  $\text{T}_{\text{E}}\text{X}$ em související, problém. Naopak, tato brožura nemůže být chápána jako podrobný návod jak pracovat s  $\text{T}_{\text{E}}\text{X}$ em pro ty, kteří ještě nemají s tímto systémem žádné zkušenosti. Těm je v brožuře věnován pro motivaci krátký úvod, v kterém je shrnuta historie a stručná charakteristika  $\text{T}_{\text{E}}\text{X}$ u.*

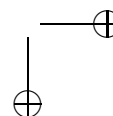
*Dokument vznikl úpravou z „Začínáme s  $\text{T}_{\text{E}}\text{X}$ em na [fi | studovna].muni.cz“, děkujeme tímto jeho autorům za poskytnutí zdrojového kódu.*

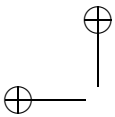




## Rejstřík

- AucT<sub>E</sub>X, 7
- bakalářská práce, 15
- Balík
  - graphics, 14
  - graphicx, 14
  - hyperref, 15
- BibT<sub>E</sub>X, 6, 11–12
- citace, viz BibT<sub>E</sub>X
- CSIndex, viz MakeIndex
- ctocs, 8
- CS TUG, 6
- čeština, viz Kódování
- diplomová práce, 15
- dvips, 8, 14
- Editory
  - emacs, 7
  - joe, 7
  - pico, 7
  - vim, 7
- hypertext, 15
- Kódování, 7
  - ISO-8859-2, 8
  - utf-8, 8
  - Windows 1250, 8
  - změna, 8
- L<sup>A</sup>T<sub>E</sub>X
  - struktura dokumentu, 8
  - úvod, 5–6
- literatura, 4, 17–18
- MakeIndex, 6, 10–11
- METAFONT, 13–14
- METAPOST, 13
- obrázky, 13–14
  - .eps, 14
  - .jpg, 14
  - .pdf, 14
  - .tif, 14
- packages, 9
- pdfT<sub>E</sub>X, 8, 14
- picture, 13
- plainT<sub>E</sub>X, 4, 5
- PostScript, 15
- Prohlížeče
  - Acrobat Reader, 8
  - GhostView, 8
  - Windvi, 8
  - xdvi, 8
- předložky, viz vlna
- rejstříky, viz MakeIndex
- TBN, 17
- T<sub>E</sub>X Live 2003, 6
- tisk, 15
- TST, 17
- vlna, 10
- xpdf, 8
- xterm, 7
- Zpravodaj CS TUG, 18





## 1 Co je to $\TeX$ ?

$\TeX$  (čte se tech) je programový systém pro elektronickou sazbu, jehož autorem je prof. Donald E. Knuth ze Stanfordské univerzity. První verze  $\TeX$ u, která byla dána k dispozici veřejnosti, byla dokončena v roce 1982.

Práci v  $\TeX$ u bychom mohli přirovnat k programování v některém z programovacích jazyků. Nejprve vytvoříte zdrojový text (k tomuto účelu vám poslouží kterýkoliv ASCII editor), přeložíte jej  $\TeX$ em, ten vám případně vypíše chybové hlášky, a poté, co chyby opravíte, získáte přeložený dokument ve formátu DVI.  $\TeX$  jako takový obsahuje zhruba 300 základních značkovacích příkazů. Pomocí makrojazyka, který je součástí  $\TeX$ u, můžete vytvářet nové sázecí příkazy složitějšími konstrukcemi z příkazů základních. Díky tomu si můžete vytvořit prostředí tak, aby vyhovovalo co nejlépe vašim potřebám. Takto vznikla také spousta více či méně známých formátů  $\TeX$ u, jako je  $\LaTeX$ ,  $\AMS\TeX$ , ale i plain $\TeX$ .

K velkým přednostem  $\TeX$ u patří jeho ekonomická a strojová nenáročnost.  $\TeX$  jako takový byl prof. Knuthem dán veřejnosti zdarma. Kdokoliv jej tedy může volně využívat, případně jej různě vylepšovat nebo dokonce vytvářet s jeho využitím nové systémy, obecně je však již nesmí označovat  $\TeX$ . Existují však i implementace  $\TeX$ u, které zdarma nejsou.

Co se týče požadavků na hardware, je  $\TeX$  velmi nenáročný. Můžete jej používat i na zastaralých PC, u kterých již jiný současný software s velkou pravděpodobností vůbec nepoběží. Zjednodušeně řečeno, stejného výsledku dosáhnete jak na staříčkém počítači typu PC XT, tak na nejnovějších výkonných stanicích. Zde je také dobré uvést, že  $\TeX$  je implementován v mnoha vzájemně se dosti lišících systémech (UNIX, MS Windows, MacOS X, ...).

K dalším přednostem  $\TeX$ u patří kompatibilita jeho implementací. Ze stejného vstupu získáte vždy stejný výstup, nezávisle na použité verzi  $\TeX$ u.

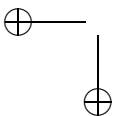
### 1.1 Zdroje informací o $\TeX$ u

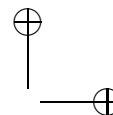
Začátečníkovi v  $\TeX$ u lze vřele doporučit např. publikaci  *$\LaTeX$  pro začátečníky* [1] nebo *Jemný úvod do  $\TeX$ u* [2] či [3]. Pro již zkušenější uživatele bude zajisté velmi užitečný manuál  *$\TeX$ book* [4] nebo kniha [5]. Mnoho informací týkajících se  $\TeX$ u naleznete také v knize *Typografický systém  $\TeX$*  [6] a  *$\TeX$ book naruby* [7].

Zmíněné publikace si lze např. objednat prostřednictvím knihkupectví Mareček (viz <http://marecek.kup.to>). Mnoho užitečného naleznete taktéž na internetové adrese  $\CS$ TUGu [8].

## 2 Co je to plain $\TeX$ ?

Plain $\TeX$ , jak již bylo uvedeno v úvodním odstavci, je makrojazykem vytvořené prostředí neboli formát  $\TeX$ u. Tento formát naprogramoval sám autor  $\TeX$ u Donald Knuth. Plain $\TeX$  je považován za základní formát, od kterého je většina dalších běžně užívaných formátů více či méně odvozena.





Pro představu, plainTeX rozšiřuje počet příkazů, které může uživatel použít, na devět set. Výhodou tohoto formátu je jeho stabilita (např. oproti L<sup>A</sup>T<sub>E</sub>Xu), a rychlost kompilace. Pro práci v tomto formátu se zpravidla rozhodují uživatelé tehdy, pokud nepotřebují psát složité dokumenty, nebo tehdy, pokud chtějí vytvářet svůj vlastní formát či prostředí a užití jiného, vyššího formátu by bylo pro ně nevhodné (příliš neprůhledné).

PlainTeX však vyžaduje schopnost programátorského myšlení, protože většinu maker si musí uživatel vytvořit a odladit sám, a to může být časově náročné.



### 3 O L<sup>A</sup>T<sub>E</sub>Xu

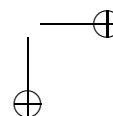
L<sup>A</sup>T<sub>E</sub>X je nadstavbou T<sub>E</sub>Xu vytvořenou panem Leslieem Lamportem. A k čemu a pro koho je L<sup>A</sup>T<sub>E</sub>X určen? Tento systém byl vytvořen zejména proto, aby zjednodušil sazbu dokumentů v T<sub>E</sub>Xu a zpřístupnil tak jinak poněkud složitý jazyk běžnému uživateli. L<sup>A</sup>T<sub>E</sub>X nabízí např. automatické číslování různých objektů, jako jsou kapitoly, odstavce, tabulky, obrázky apod. Dále umí automaticky sestavit obsah či snadno vytvářet rejstříky. Obsahuje příkazy, pomocí kterých je možné vytvářet jednoduché obrázky. Taktéž má již předvoleny velikosti nadpisů kapitol, odstavců apod. Standardní styly se však řídí americkými typografickými konvencemi, které se značně liší od těch evropských. Proto bylo vytvořeno i mnoho stylů, které respektují evropská typografická pravidla.

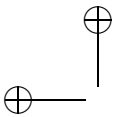
Je ovšem vhodné zdůraznit, že zdrojový soubor dokumentu vytvořený v L<sup>A</sup>T<sub>E</sub>Xu se liší od zdrojového souboru v T<sub>E</sub>Xu. Proto, pokud tento L<sup>A</sup>T<sub>E</sub>Xový soubor přeložíte T<sub>E</sub>Xem, dočkáte se z největší pravděpodobností pouze spousty chybových hlášek a přeloženého souboru, který nebude zdaleka odpovídat vašim představám. Jestliže se pokusíte naopak přeložit plainTeXový soubor programem (cs)latex, nepodaří se vám to vůbec.

Problém je v poněkud odlišné syntaxi L<sup>A</sup>T<sub>E</sub>Xu oproti základnímu T<sub>E</sub>Xu. Charakteristickým rysem vstupních textů ve formátu L<sup>A</sup>T<sub>E</sub>X jsou prostředí vymezená příkazy `\begin` a `\end`, které jste nuceni při tvorbě textu dodržovat. Tyto bloky mohou být do sebe vnořovány stejně jako např. u některých programovacích jazyků pascalovského typu. Pro názornost uvedeme příklad jednoho a téhož dokumentu vytvořeného nejprve v plain-T<sub>E</sub>Xu a poté v L<sup>A</sup>T<sub>E</sub>Xu. U obou z těchto vstupních textů vznikne po přeložení příslušnými programy (csplain a cslatex) velmi podobně vysázený dokument, ale zdrojové soubory se evidentně liší.

#### Vstupní soubor v plainTeXu:

<code>\input czech.sty</code>	Zprávu, kterou právě čtete,
<code>\nopagenumbers</code>	nečtete, nic se v~ní
	nedočtete...
<code>\rightline{</code>	
<code>V~Brně dne \today}</code>	<code>\vskip 0.6cm S~pozdravem</code>
<code>{\bf Vážený čtenáři,}</code>	<code>\rightline{\it Kdosi cosi}</code>
<code>\vskip 0.9cm</code>	<code>\bye</code>





### Vstupní soubor v $\LaTeX$ u:

```
\documentclass{article}          \vspace{0.9cm}
\usepackage{czech}              Zprávu, kterou právě čtete,
\pagestyle{empty}              nečtete, nic se v~ní
\begin{document}               nedočtete\dotscdots
                                \vspace{0.6cm}

\begin{flushright}
V~Brně dne \today              S~pozdravem
\end{flushright}              \begin{flushright}
                                \textit{Kdosi cosi}
\textbf{Vážený čtenáři,}      \end{flushright}
                                \end{document}
```

$\LaTeX$  některé příkazy obsažené v základním formátu nezná a místo nich má předdefinována prostředí (v uvedeném příkladu je to např. místo příkazu `\rightline` prostředí `flushright`). Jiným příkladem nekompatibility  $\LaTeX$ u vzhledem k `plainTeX`u jsou příkazy, které mají v obou formátech stejný název, ale různou funkci (např. `\line`). Avšak  $\TeX$ ová primitiva lze samozřejmě v  $\LaTeX$ u použít.

Na tomto místě je také vhodné upozornit na skutečnost, že formát  $\LaTeX$  se stále vyvíjí. Původní verze  $\LaTeX$ u označovaná jako verze 2.09 již není ani autorem ani  $\LaTeX$ -tým nadále podporována a udržována. Na místo toho vznikl a rozšířil se  $\LaTeX 2_{\epsilon}$  (pracuje se na verzi označované  $\LaTeX 3$ ).

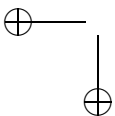
K formátu  $\LaTeX$  existuje manuál L. Lamporta  *$\LaTeX$ —A Document Preparation System* [9]. Velmi užitečná je také série knih *The  $\LaTeX$  Companion* [10, 11, 12]. Z českých psaných materiálů je velmi vhodnou příručkou pro začínající uživatele  $\LaTeX$ u výše uvedená kniha  *$\LaTeX$  pro začátečníky* [1]. Doporučujeme také navštívit webové stránky  $\mathcal{C}\mathcal{S}\mathcal{TUG}$ u [8].

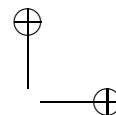
## 4 $\TeX$ pod UNIXem

Uživatelům na matematice je v současnosti k dispozici  $\TeX$  na serverech queen (pro zaměstnance) a bart (pro studenty). Zde  $\TeX$  běží pod operačním systémem SUSE Linux.

### 4.1 Co to je $\TeX$ Live 2003?

$\TeX$  Live 2003 je distribuce  $\TeX$ u pro UNIX a Windows32 systémy. Zahrnuje  $\TeX$ ,  $\LaTeX 2_{\epsilon}$ , METAFONT, METAPOST, MakeIndex a Bib $\TeX$ , velké množství maker, fontů a dokumentace. Instalační DVD nebo CD  $\TeX$  Live 2003 obsahuje pouze volně šířený software a je možné si je zapůjčit na LVT sekce matematika nebo získat členstvím v  $\mathcal{C}\mathcal{S}\mathcal{TUG}$ u. Dokonce je možné spouštět programy přímo z CD/DVD bez nutnosti instalace na disk. Domovská stránka projektu je <http://tug.org/texlive/>, kompletní dokumentace je k dispozici v [13].





## 5 Editory

Pro vytváření  $\text{T}_{\text{E}}\text{X}$ ovských dokumentů můžete využít řadu editorů. V podstatě jediným omezením, co se týče možnosti využít ten či onen editor pro psaní  $\text{T}_{\text{E}}\text{X}$ ovských dokumentů, je to, aby tento editor byl schopen textový soubor ukládat v čisté formě, tedy bez svých řídicích znaků. Zejména na unixových strojích je to např. základní editor `vim` (příp. `vi`). Díky svému poněkud odlišnému ovládání oproti jiným zejména neunixovým editorům může `vim` dělat začátečníkovi při vytváření dokumentů určité problémy. Proto je nezbytné se seznámit alespoň se základními prvky ovládání tohoto editoru, a to např. přímo příkazem `:help` v editoru `vim`, na manuálové stránce (`man vim`) nebo v příručkách o Unixu. Mezi další editory, které jsou vám na unixových strojích k dispozici, patří `pico` nebo `joe`.



### 5.1 Emacs / Auc $\text{T}_{\text{E}}\text{X}$

Asi nejlépe uzpůsobeným nástrojem pro tvorbu nejen  $\text{T}_{\text{E}}\text{X}$ ovských dokumentů, který můžete na sekčních strojích používat, je editor `emacs`.

Emacs nabízí mnoho konfiguračních možností, naprogramovaných v jazyce Lisp, včetně podpory pro psaní textů v  $\text{T}_{\text{E}}\text{X}$ u. Můžete z něj přímo volat  $\text{T}_{\text{E}}\text{X}$  a různé obslužné programy, jednoduše pomocí konkrétních sekvencí vkládat do textu některé časté  $\text{T}_{\text{E}}\text{X}$ ovské konstrukce apod. Kromě toho můžete také využít balík lispovských marker pro Emacs – Auc $\text{T}_{\text{E}}\text{X}$ , který nabízí integrované prostředí pro tvorbu textů zejména v  $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ u.

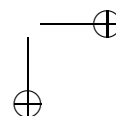
Vzhledem k tomu, jaké možnosti Emacs skýtá, má také přiměřeně obsáhlou dokumentaci (zhruba 300 stran uživatelského manuálu + 700stránkový manuál pro psaní lispovských kódů). Je také třeba upozornit, že stejně jako u `vim` editoru, může být pro začátečníky poněkud problematické si zvyknout na ovládání a vůbec jiné prostředí Emacsu ve srovnání s DOSovskými editory. Kdo se však nenechá odradit těmito zdánlivými nevýhodami a rozhodne se Emacs používat, určitě nebude litovat :-).

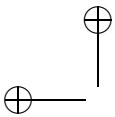
## 6 Kódování, čeština a spol.

Velmi důležité pro tvorbu česky psaných  $\text{T}_{\text{E}}\text{X}$ ovských dokumentů je „nastavení češtiny“. Toto můžeme rozdělit na dvě oblasti: jednak je důležité, aby se vám korektně zobrazovaly české znaky při tvorbě dokumentu, tedy v terminálovém okně resp. v prostředí některého z editorů.

Pokud váš `xterm` píše místo českých znaků číslíce, stiskněte současně klávesy levý Shift a Alt (případně Scroll Lock).

Další důležitou věcí je kódování znaků vašeho vstupního  $\text{T}_{\text{E}}\text{X}$ ovského souboru. Zejména je to aktuální, jestliže si přinášíte již rozpracovaný dokument odjinud. Může se stát, že takový text bude v jiném kódování, než jaké je běžně nastaveno na unixových strojích, tedy ISO-8859-2. Toto mívá někdy za následek špatně zobrazené české znaky v přeloženém výstupním souboru. Pozor! Např. pokud přeložíte donesený  $\text{T}_{\text{E}}\text{X}$ t, který je kódovaný ve Windows 1250, přeložený dokument se vám bude možná zdát na první pohled v pořádku. Ona se totiž většina českých znaků zobrazuje stejně jak ve Windows





1250, tak v ISO-8859-2. Neplatí to ale např. pro „ž“, a tak bude váš přeložený dokument prošpikován znaky „l“ . . .

Abyste se vyhnuli těmto problémům, měli byste vstupní T<sub>E</sub>Xty, které jsou uloženy v jiném kódování (včetně utf-8), převést do ISO-8859-2, což je implicitní vstupní kódování T<sub>E</sub>Xu na unixových strojích na matematice (ve Windows je to pak kódování Windows 1250).

## 6.1 Jak převádět texty mezi různými kódováními

Na serverech je nainstalován program `cstocs`, který umí převádět mezi jednotlivými kódováními češtiny.

Např. pokud budete chtít převést dokument `priklad.tex` v kódování Windows 1250 do kýženého kódování ISO-8859-2, budete postupovat takto:

Spustíte

```
cstocs 1250 il2 <priklad.tex >priklad.il2.tex
a v souboru priklad.il2.tex máte váš soubor v kódování ISO-8859-2. Pro úpl-
nost zde uvedeme, s kterými kódováními program cstocs verze 3.4 umí pracovat:
    1250 1252 ascii cork ill il2 kam koi8 mac macce
        pc2 pc2a tex utf8 vga
```

Podrobnosti naleznete na manuálové stránce (viz příkaz `man cstocs`).

## 7 Prohlížeče a konverze mezi formáty

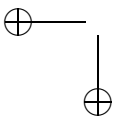
Při práci s T<sub>E</sub>Xem se většinou setkáte s několika různými formáty, ve kterých bude váš dokument přeložen. Jakmile přeložíte dokument programem `(la)tex`, získáte `.dvi` soubor. Na počítači `bart` je vám k dispozici pro unixové platformy program `xdvi`, který vám umožní tento soubor si prohlédnout. Obdobou tohoto prohlížeče pro windowsovské stroje je program `windvi`.

Jestliže chcete získat dokument ve formátu PostScript, můžete použít pro převod z `dvi` formátu program `dvips`. Získáte tak `.ps` soubor. Zobrazit si jej můžete v unixu pomocí programu `Ghostview` (příkazem `gv`, pod Windows pak příkazem `gsview32`). Do PDF formátu je možno převést dokument z PostScriptu např. pomocí programu `pstopdf`. Další možnost, jak `.pdf` soubor získat, je využít přímo program `pdftex` (`pdflatex`, `pdfcslatex`, . . .), který vám jej vytvoří ze vstupního souboru (L<sup>A</sup>)T<sub>E</sub>Xu [14]. Pro prohlédnutí získaného `.pdf` souboru lze využít program `Acrobat Reader`, `Ghostview` nebo na některých unixových platformách program `xpdf`.

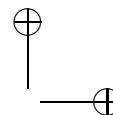
## 8 . . . a jdeme (L<sup>A</sup>)T<sub>E</sub>Xovat . . .

Pro přehlednost nejprve uvedeme, jakou má dokument určený ke zpracování systémem L<sup>A</sup>T<sub>E</sub>X strukturu.

```
\documentclass[volby]{třída}[datum vytvoření]
.
. preamble
```







```
.
\begin{document}
.
. textová část
.
\end{document}
```

Příkaz `\documentclass` spolu s parametrem *třída*, který určuje styl sazby dokumentu, je povinný. Ze standardních tříd máte na výběr `article` (pro článek), `book` (pro knihu), `letter` (pro dopis), `report` (pro zprávu) a `slides` (pro průsvitné fólie). Příslušné textové soubory s příponou `.cls`, v kterých jsou uloženy definice tříd, můžete samozřejmě libovolně upravovat, a vytvářet si tak své vlastní třídy uzpůsobené pro vaše vlastní použití.

Další, volitelný, parametr příkazu `\documentclass` vám umožňuje modifikovat činnost příkazů ve zvolené třídě. Uvést můžete i několik voleb najednou, musíte je však oddělit čárkami bez mezer, např.

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

V preambuli dokumentu uvedené příkazy mají platnost pro celý text. V této části lze k dokumentu připojit další balíky příkazů (packages), a to pomocí příkazu

```
\usepackage[volby]{balík}[datum vytvoření].
```

Význam a struktura parametrů tohoto příkazu jsou prakticky stejné, jako u příkazu `\documentclass`.

Jakmile některý z balíků takto připojíte do preambule dokumentu, můžete příkazy v balíku nadefinované používat v celém dokumentu.

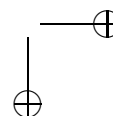


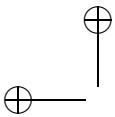
## 9 Balík pro psaní českých dokumentů (`czech.sty`)

Pro tvorbu česky psaných dokumentů je určen balík `czech.sty`. Po jeho zavolání (např. v  $\LaTeX$ u přidáním `\usepackage{czech}` do preambule dokumentu) se změní činnost  $(\LaTeX)$ u v několika bodech:

- Zapne se české dělení slov. Znamená to, že bude  $(\LaTeX)$  pracovat s českou tabulkou dělení slov a tabulkou výjimek, přičemž tabulku výjimek lze postupně rozšiřovat na libovolných místech dokumentu, a to pomocí příkazu `\hyphenation`.
- Mezery za tečkami a čárkami budou menší, odpovídající lépe normám české sazby.
- Je definován příkaz `\uv{...}` pro psaní českých uvozovek.
- Předefinuje se příkaz `\chapter` tak, že způsobí sazbu slova „Kapitola“ místo původně definovaného slova „Chapter“. Stejně tak se „počeští“ další automaticky generovaná slova, jako např. „Literatura“, „Obrázek“ apod.
- Příkaz `\today` vysází datum česky.

Dalším nástrojem pro psaní vícejazyčných dokumentů včetně rudimentární podpory češtiny je standardní balík  $\LaTeX$ u zvaný přízračně Babel.





## 10 Program vlna

Řádky v textu mohou být děleny buď „uvnitř“ slova (o tom jsme se zmínili výše), a nebo v místě mezer.  $\TeX$ u je možné zakázat, aby v určité mezeře řádek rozdělil. Provede se to znakem “~”. Např. je nesprávné z hlediska typografické normy, aby se na konci řádku vyskytovaly neslabičné předložky. V takových případech znak “~” lze účelně využít (Např.  $V\sim\TeX\{u\}$  využíváme vlnku...).

Při psaní textu však není třeba, co se týče neslabičných předložek, na vkládání „vlnek“ stále pamatovat. Místo toho stačí kdykoliv provést jejich doplnění automaticky, a to programem vlna.

Bohužel však existuje mnoho dalších míst, kde podle pravidel sazby nelze řádky rozdělit a kde vlna „vlnku“ nedoplní (např. za iniciálami:  $J.\sim K.\sim Ty1$  apod.). Na taková místa je potřeba dávat při psaní dokumentu pozor.

## 11 Rejstříky a citace

Pro snazší tvorbu rejstříků a seznamů citací jsou vám k dispozici např. speciální programy MakeIndex ( $\mathcal{C}\mathcal{S}$ Index) a Bib $\TeX$ .

### 11.1 MakeIndex

Pokud chcete, aby váš dokument obsahoval také rejstřík, musíte v textu konkrétním způsobem označit ta hesla, která se v rejstříku mají objevit. Provádí se to pomocí příkazu  $\backslash\text{index}$ , za jehož argument uvedete heslo spolu s příp. dalšími nastaveními (podhesla, namísto čísel stránek zobrazovat stránkový rozsah apod.). Tyto příkazy se vkládají přímo do míst v textu, na které má příslušné heslo rejstříku odkazovat.

Např.:

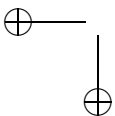
Těleso ponořené do kapaliny  
 $\backslash\text{index}\{\text{těleso!ponořené do kapaliny}\}$  je nadlehčováno silou,  
která se rovná tíze  $\backslash\text{index}\{\text{tíha}\}$  kapaliny tělesem vytlačené.

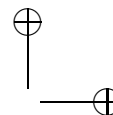
Tento příklad nám v konečné fázi vysází rejstřík takto:

těleso  
    ponořené do kapaliny, 3  
tíha, 5

Nejprve je třeba přidat do preambule dokumentu příkaz  $\backslash\text{makeindex}$ . Ten způsobí, že se všechny příkazy  $\backslash\text{index}$  uvedené v textu vypíší do souboru s příponou  $.idx$ . V daném souboru tak vznikne jakýsi polotovar budoucího rejstříku – nesetříděný, neupravený do tvaru seznamu. Z tohoto  $.idx$  souboru vytvoří plnohodnotný rejstřík program MakeIndex nebo jeho česká verze  $\mathcal{C}\mathcal{S}$ Index. K jejich použití je potřeba připojit do preambule dokumentu balík makeidx příkazem  $\backslash\text{usepackage}\{\text{makeidx}\}$ .

Nyní už jen vložíte na místo v textu, kde chcete, aby byl rejstřík vysázen, příkaz  $\backslash\text{printindex}$ .





Vzniklý dokument přeložíte (např. `cslatex priklad`), čímž se vytvoří zmiňovaný soubor s příponou `.idx`. Poté spustíte program `makeindex` nebo `csindex` spolu se jménem tohoto souboru (`csindex priklad`). Tím se vytvoří již setříděný rejstřík, který se uloží do souboru s příponou `.ind`. Nakonec dokument ještě jednou přeložíte, přičemž při tomto druhém překladu již příkaz `\printindex` (viz výše) nalezne potřebný `.ind` soubor a načte jej. Tím vytvoří na požadovaném místě utříděný rejstřík. Způsobů, jak vytvořit rejstřík, je pochopitelně víc. Např. lze využít alternativního programu `xindy`.

## 11.2 BibT<sub>E</sub>X

Na tomto místě se krátce zmíníme také o možnostech automatického vytváření seznamů citací. K tomuto účelu, zejména pokud tvoříte ve formátu L<sup>A</sup>T<sub>E</sub>X, je vám k dispozici program BibT<sub>E</sub>X. Nyní si stručně ukážeme jak jednoduše vytvořit seznam citací pomocí tohoto programu.

Nejprve si vytvoříte databázi publikací, a to v souboru s příponou `.bib` (např. `citace.bib`). Struktura tohoto souboru musí dodržovat přesně danou syntax. Pro představu uvedeme příklad, jak může taková databáze vypadat.

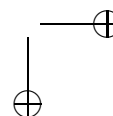
```
@ARTICLE{honza,  
author = "Novák, Jan",  
title = "Můj život",  
journal = "Životopisy",  
number = 4,  
year = 1997,  
pages = "14--16",  
month = "duben",  
}  
  
@BOOK{karel,  
author = "Nikdo, Karel",  
title = "Kniha o {A}rchimédovi",  
publisher = "Nakladatelství Ťuk",  
address = "Brno, Nějaká 15",  
note = "První vydání",  
year = "1992",  
}
```

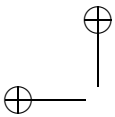


Jde vlastně o soubor záznamů určitého typu (v tomto případě typu `@ARTICLE` a `@BOOK`), které obsahují jednotlivé údaje. Každý typ má svou vlastní skupinu povinných údajů.

Na příkladě vidíme, že každý záznam obsahuje mimo údajů také pracovní značku (zde je to `honza` resp. `karel`). Tu použijete v dokumentu spolu s příkazem `\cite` (např. `\cite{karel}`) tam, kde budete chtít na tu či onu publikaci (záznam) ze seznamu citací odkázat.

Zde je vhodné upozornit na skutečnost, že v textech názvů publikací (to, co je v položce `title=`) se všechna písmena kromě prvního vysázejí automaticky jako malá, bez





ohledu na to, jak jsou napsána v souboru vaší databáze. Pokud chcete vysázet název s velkými písmeny, musíte je uzavřít do složených závorek (viz předchozí příklad). Pro to, aby se v dokumentu seznam použitých citací vysázel, je dále potřeba do vstupního textu dokumentu mimo preambuli uvést příkaz `\bibliography{citace}`. Dále uvedeme příkaz `\bibliographystyle{jmeno.styl}`, který specifikuje styl použitý programem BibTeX. Ze základních stylů jsou to např. `plain.bst`, `alpha.bst`, `abbrv.bst` a `unsrt.bst`. Jednotlivé styly se od sebe navzájem liší ve formátu výstupního souboru, ale většinou pouze nepatrně. Např. styl `unsrt.bst` způsobí, že výstup nebude uspořádán podle abecedy, zatímco styl `plain.bst` BibTeXu přikáže, aby výstup uspořádal.

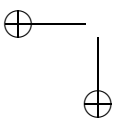
Dokument i databázi máme nyní připravenou. A co dál?

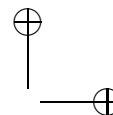
- Dokument přeložíme L<sup>A</sup>T<sub>E</sub>Xem, čímž se vytvoří soubor s příponou `.aux`, který bude obsahovat instrukce (jméno souboru s databází, použitý styl apod.) pro program BibTeX.
- Spustíme program `bibtex` spolu se jménem dokumentu resp. souboru `.aux`, tedy např. `bibtex priklad`. BibTeX vytvoří automaticky podle instrukcí `.bbl` soubor se seznamem citací ve formátu, který je zpracovatelný L<sup>A</sup>T<sub>E</sub>Xem.
- Opět přeložíme náš dokument příkazem `cslatex`. Tím se načte do dokumentu již vytvořený `.bbl` soubor.
- Ještě jednou dokument přeložíme, čímž se zbavíme varovných hlášek o nepřiznačených značkách, která se nám u předchozích zpracování L<sup>A</sup>T<sub>E</sub>Xem objevovala (kvůli souboru `.bbl`). A jsme hotovi.

Rejstřík, jakož i seznam literatury, se standardně sází na zvláštní stránku. To ovšem nemusí být vždy žádoucí. Někdy je potřeba např. z důvodu úspory místa vysázet rejstřík tak, aby navazoval na konec textu a sázel se do dvou sloupců. K tomu je třeba modifikovat okolí `theindex` ze stylu dokumentu, který používáte. Soubor s definicemi stylu můžete vyhledat využitím programu `kpsewhich` (např. spustíte `kpsewhich book.cls`). Zkopírujete si z něj definici okolí `theindex` do preambule vašeho dokumentu mezi závorky `\makeatletter` a `\makeatother`, a poté nadefinujete novou verzi okolí příkazem `\renewenvironment`. Zejména se zaměřte na příkaz `\twocolumn`, nebo použijte (doporučeno) balík `multicol`.

V dokumentu využívajícím třídu `article` by tato úprava mohla vypadat například takto:

```
\usepackage{multicol}
\makeatletter          % Umožňuje použít @ v názvech maker
\renewenvironment{theindex}
{\section*{\indexname}%
 \begin{multicols}{2}
   \@mkboth{\MakeUppercase\indexname}%
   {\MakeUppercase\indexname}%
 \thispagestyle{plain}\parindent\z@
```





```
\parskip\z@ \@plus .3\p@\relax
\let\item\@idxitem}
{\end{multicols}}
\makeatother          % Zakazuje použít @ v názvech maker
```

Řešení pro bibliografii (okolí thebibliography) je analogické.

## 12 Práce s obrázky

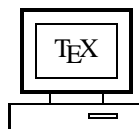
Obrázky můžete do dokumentu vkládat různými způsoby. Zejména záleží na tom, zda jde o obrázky rastrové nebo vektorové.

$\TeX$  jako takový nebyl navržen pro tvorbu nebo vkládání obrázků. Přesto existují makra, která vám umožní jednoduché obrázky sestavovat. Vektorové obrázky jste tak schopni vytvářet v omezené míře např. přímo v  $\LaTeX$ u v prostředí `picture`. Druhou možností je využít systému METAFONT nebo METAPOST. Rastrové, ale i vektorové obrázky, které byly vytvořeny některým jiným specializovaným programem, se do textu vloží buď převedením na písmena znakové sady, nebo využitím vlastností ovladače pro zobrazení souboru `.dvi`.

### 12.1 Prostředí `picture`

$\LaTeX$ ové prostředí `picture` je uzpůsobeno pro tvorbu spíše jednoduchých schémat a obrázků. Podrobněji se s tímto prostředím můžete obeznámit např. v knize  *$\LaTeX$  pro začátečníky* [1]. Pro představu uvedeme příklad obrázku:

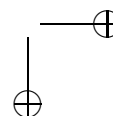
```
\begin{picture} (35,55)
\linethickness{1pt}
\put(5,15){\framebox(40,30){\TeX}}
\thinlines
\put(10,20){\framebox(30,20){}}
\put(15,15){\line(0,-1){5}}
\put(35,15){\line(0,-1){5}}
\put(0,0){\framebox(50,10){}}
\put(30,5){\framebox(10,1){}}
\end{picture}
```

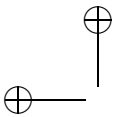


Jak je vidět, vytváření obrázků touto cestou není příliš pohodlné, zejména, pokud tvoříme obrázky složené z více objektů. Jen pro zajímavost existuje program `xfig`, ve kterém můžete obrázky vytvářet v grafickém prostředí pomocí myši a po uložení získáte textový soubor obsahující `\begin{picture} ... \end{picture}` vašeho obrázku.

### 12.2 Krátce o METAFONTu a METAPOSTu

Pokud však chcete vytvářet složitější vektorové obrázky, které se nebudou skládat pouze ze základních grafických elementů, jako je tomu u  $\LaTeX$ ového prostředí `picture`,





můžete využít schopností programu METAFONT. Tento program určený k návrhu a realizaci písem je možné logicky využít i pro vytváření jiných obrázků než je návrh písma. Zmiňme se nyní také krátce o METAPOSTu. Je to programovací jazyk, vzniklý z jazyka METAFONT, určený pro popis obrázků. Hlavním rozdílem mezi METAFONTEM a METAPOSTem je jejich rozdílný výstup. Zatímco u METAFONTu je výstupem bitová mapa a metrika, u METAPOSTu je to program v jazyce PostScript. Vzhledem k tomu, že problematika tvorby obrázků v METAFONTu je značně obsáhlá a rozebírat ji zde by zabralo příliš mnoho místa, nezbyvá než odkázat na příslušnou literaturu.

### Zdroje informací o METAFONTu

Jako úvod do METAFONTu může posloužit *Zpravodaj C<sub>S</sub>TUG* [15] 1/1998, který můžete buď zakoupit v knihkupectví Mareček, příp. stáhnout v elektronické podobě z adresy <http://bulletin.cstug.cz>.

Cenné informace můžete získat také z dalších čísel *Zpravodaje* (např. 3/1991, 2/1992, 4/1994, 3–4/1998 aj.) nebo z webových stránek C<sub>S</sub>TUGu [8].

Stejně jako *The T<sub>E</sub>Xbook* [4] k T<sub>E</sub>Xu, sestavil D.E. Knuth také manuál k METAFONTu – *The METAFONTbook* [16].

Vcelku podrobně se METAFONTEM zabývá taktéž již výše zmiňovaná kniha *Typografický systém T<sub>E</sub>X* [6].

## 12.3 Balíky `graphics` a `graphicx`

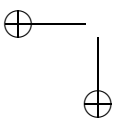
Další možností, jak pracovat s obrázky v L<sup>A</sup>T<sub>E</sub>Xu, je využít možností standardního balíku `graphics` nebo raději balíku `graphicx`. U těchto balíků se předpokládá existence ovladače, který je schopen pracovat s obrázky (standardně se jedná o zpracování grafiky zapsané v jazyce PostScript). Podrobněji se s těmito balíky můžete seznámit např. ve skriptech *Sázíme v L<sup>A</sup>T<sub>E</sub>Xu diplomovou práci z matematiky* [17].

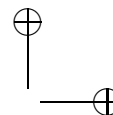
Pro vložení PostScriptového obrázku (zpravidla má příponu `.eps`) je definován příkaz `\includegraphics{jméno_souboru}`.

Balík `graphics` vyžaduje, aby byl výstup v jazyce PostScript. Proto musí být obvyklým překladem získaný výstupní `.dvi` soubor převeden do PostScriptového souboru (např. pomocí programu `dvips`).

Pokud byste však chtěli vkládat PostScriptové obrázky (`.eps`) pomocí programu `pdflatex`, neuspějete. Ten totiž v sobě neobsahuje interpret jazyka PostScript. Musíte tedy daný obrázek nejprve zkonvertovat do `.pdf` (příp. `.jpg`, `.tif` nebo `.png`). To lze provést např. programem `epstopdf` nebo pomocí programu `convert`. Takto zkonvertovaný obrázek již můžete vložit do dokumentu standardně pomocí příkazu `\includegraphics`. Je však potřeba v preambuli dokumentu, kde voláte příslušný balík `graphics`, uvést navíc parametr `pdftex`, tedy `\usepackage[pdftex]{graphics}`.

Více podrobností nejen o vkládání obrázků pomocí programu `pdflatex` naleznete např. na stránkách *PdfT<sub>E</sub>X Support* [14]. Podrobně se problematice vkládání externí grafiky věnuje také např. příručka *Using Imported Graphics in L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>* [18].





## 13 Jak vložit do dokumentu logo?

Při psaní dokumentu bývá občas potřeba vložit do textu nějaké logo. Zde si uvedeme příklady některých z nich:

**T<sub>E</sub>X** resp. **L<sup>A</sup>T<sub>E</sub>X** — příkazem `\TeX` resp. `\LaTeX`

**METAFont** resp. **METAPOST** — po přidání balíku `mfllogo` příkazem `\MF` resp. `\MP`

**C<sub>S</sub>Index**, **C<sub>S</sub>TUG** apod.

```

\newcommand{\CS}{%
  \mbox{$\cal C\kern-.1667em\lower.5ex\hbox{%
    $\cal S$}\kern-.075em $}}
\newcommand{\cstug}{\CS\thinspace TUG}
\newcommand{\csindex}{\CS\thinspace index}

```

## 14 Tvorba diplomových a bakalářských prací

O obecných zákonitostech sazby diplomových prací pojednává např. dokument na adrese <http://home.pf.jcu.cz/~edpo/pravidla/pravidla.html>.

Pokud máte zájem vytvářet také elektronickou podobu vaší práce obohacenou o hypertextové odkazy, můžete k tomuto účelu využít např. balík `hyperref`. Podrobné informace o tom, jak s tímto balíkem pracovat, lze nalézt např. na adrese <http://www.tug.org/applications/hyperref/> [19].

Další informace vztahující se k tvorbě hypertextových dokumentů je možné mimo jiné nalézt taktéž v CSFAQ na stránkách C<sub>S</sub>TUGu [8].



## 15 Fonty MathTime

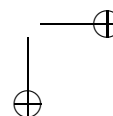
Jedná se o komerční fonty pro sazbu matematiky, které sekce zakoupila a jsou k dispozici na počítačích bart a queen. Pro jejich použití je třeba do preambule dokumentu umístit příkaz `\usepackage[mtbold,mtplusscr,mtpluscal]{mathtime}`. Dává se až za `\usepackage{czech}`, ale před `\usepackage{amsmath}`.

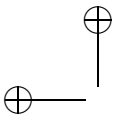
## 16 Jak vytisknout v PostScriptu více stránek na jednu A4, tisk brožury apod.

K tomuto účelu lze použít možnosti balíku `psutils`. Informace o tomto balíku včetně odpovědí na otázky podobného typu naleznete na adrese

<http://www.abclinuxu.cz/clanky/show/55273> nebo také

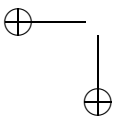
[http://www.grafika.cz/art/pdf/ps\\_utils.html](http://www.grafika.cz/art/pdf/ps_utils.html).



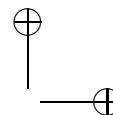


## Zpětná reakce: dotazy a připomínky

Pro dotazy týkající se konkrétních problémů se sazbou v  $\text{T}_{\text{E}}\text{X}$  použijte elektronickou konferenci `cstex@cs.felk.cvut.cz`, skupinu News `news:cz.comp.cstex` nebo FAQ na <http://www.fi.muni.cz/cstug/csfaq/>.

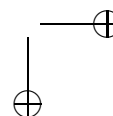


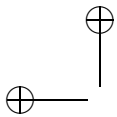




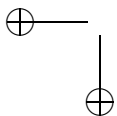
## Literatura

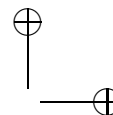
- [1] J. Rybička. *LT<sub>E</sub>X pro začátečníky – 3. vydání*. Brno: Konvoj, 2003.  
Příručka vhodná pro úplné začátečníky v L<sup>A</sup>T<sub>E</sub>Xu.
- [2] M. Doob. *Jemný úvod do T<sub>E</sub>Xu*. Praha: C<sub>S</sub>TUG, 1992.  
Manuál pro samostatné studium.
- [3] P.W. Abrahams, K. Berry, a K.A. Hargreaves. *T<sub>E</sub>X for the Impatient*. Addison-Wesley, 2003.  
Kniha, dříve vydaná renomovaným nakladatelstvím, a nyní dostupná pod GNU Free Document Licence na <http://tug.org/ftp/tex/impatient/> ve zdrojové podobě.
- [4] D. Knuth. *The T<sub>E</sub>Xbook*. Massachusetts: Addison-Wesley, 1994.  
Původní příručka k T<sub>E</sub>Xu přímo od D. Knutha.
- [5] V. Eijkhout. *T<sub>E</sub>X by Topic*. Addison-Wesley, 1992.  
Přehledná tematicky tříděná příručka pro T<sub>E</sub>Xového programátora. Volně dostupná v elektronické podobě na <http://www.eijkhout.net/tbt/>.
- [6] P. Olšák. *Typografický systém T<sub>E</sub>X*. Brno: Konvoj, 1997.  
Vhodný průvodce jak pro pokročilé uživatele T<sub>E</sub>Xu, tak i částečně pro začátečníky. Tato kniha se zabývá zejména technickými souvislostmi T<sub>E</sub>Xu a doprovodného softwaru. Často se pro tuto publikaci používá zkratky TST.
- [7] P. Olšák. *T<sub>E</sub>Xbook naruby*. Brno: Konvoj, 1997.  
Tato publikace mimo jiné podrobně rozebírá všechny algoritmy T<sub>E</sub>Xu a většinu maker plainu a cspplainu. Je vhodná zejména pro pokročilejší uživatele, kteří chtějí proniknout hlouběji do mechanismů fungování T<sub>E</sub>Xu. Pro odkazy je používána zkratka TBN.
- [8] Internetové stránky Československého sdružení uživatelů T<sub>E</sub>Xu (C<sub>S</sub>TUG). <http://www.cstug.cz>.  
Na těchto stránkách je možno nalézt jednak odkazy související s T<sub>E</sub>Xem (např. diskusní fóra, FAQ, elektronické publikace a archívy softwaru), dále pak podrobné informace o náplni C<sub>S</sub>TUGu, stanovy, výhody členství v tomto sdružení apod.
- [9] L. Lamport. *LT<sub>E</sub>X—A Document Preparation System—Users’s Guide and Reference Manual*. Massachusetts: Addison-Wesley, 1994.  
Původní manuál k systému L<sup>A</sup>T<sub>E</sub>X.
- [10] F. Goosens, M. Mittelbach. *The L<sup>A</sup>T<sub>E</sub>X Companion*. Massachusetts: Addison-Wesley, 1994.
- [11] F. Goosens, M. Mittelbach. *The L<sup>A</sup>T<sub>E</sub>X Graphics Companion*. Massachusetts: Addison-Wesley, 1997.
- [12] F. Goosens, M. Mittelbach. *The L<sup>A</sup>T<sub>E</sub>X Web Companion*. Massachusetts: Addison-Wesley, 1999.
- [13] Příručka T<sub>E</sub>Xlive.  
Několik jazykových mutací včetně české najdete v `/packages/share/texlive2003/texmf/doc/tldoc/`, 2003.





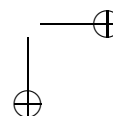
- [14] PdfTeX Support. <http://pdftex.org/>.  
Stránky o programu PdfTeX a vše co s ním souvisí.
- [15] *Zpravodaj CSTUGu*.  
Tištěný čtvrtletník distribuovaný zdarma členům sdružení. Po uplynutí 12 měsíců od tištěného vydání je poskytován v elektronické podobě (PDF) na <http://bulletin.cstug.cz>.
- [16] D. Knuth. *The METAFONTbook*. Massachusetts: Addison-Wesley, 1986.  
Původní příručka k METAFONTu přímo od jeho autora Donalda Ervina Knutha.
- [17] L. Čechová a R. Plch. *Sázíme v L<sup>A</sup>T<sub>E</sub>Xu diplomovou práci z matematiky*. MU Brno, 2003.  
Skriptum PřF MU Brno věnované především sazbě matematiky a práci s grafikou.
- [18] *Using Imported Graphics in L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>*.  
<ftp://ftp.cstug.cz/pub/tex/CTAN/info/epslatex.pdf>.  
Příručka, která se podrobně věnuje problematice vkládání externí grafiky v L<sup>A</sup>T<sub>E</sub>Xu.
- [19] *Hypertext marks in L<sup>A</sup>T<sub>E</sub>X*. <http://tug.org/applications/hyperref/manual.html>.  
Manuál k vytváření klikacích (hypertextových) dokumentů.

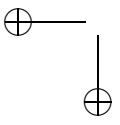
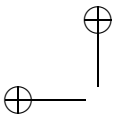


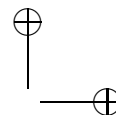


## Obsah

<b>1 Co je to T<sub>E</sub>X?</b>	<b>4</b>
1.1 Zdroje informací o T <sub>E</sub> Xu . . . . .	4
<b>2 Co je to plainT<sub>E</sub>X?</b>	<b>4</b>
<b>3 O L<sup>A</sup>T<sub>E</sub>Xu</b>	<b>5</b>
<b>4 T<sub>E</sub>X pod UNIXem</b>	<b>6</b>
4.1 Co to je T <sub>E</sub> X Live 2003? . . . . .	6
<b>5 Editory</b>	<b>7</b>
5.1 Emacs / AucT <sub>E</sub> X . . . . .	7
<b>6 Kódování, čeština a spol.</b>	<b>7</b>
6.1 Jak převádět texty mezi různými kódováními . . . . .	8
<b>7 Prohlížeče a konverze mezi formáty</b>	<b>8</b>
<b>8 ... a jdeme (L<sup>A</sup>)T<sub>E</sub>Xovat...</b>	<b>8</b>
<b>9 Balík pro psaní českých dokumentů (czech.sty)</b>	<b>9</b>
<b>10 Program vlna</b>	<b>10</b>
<b>11 Rejstříky a citace</b>	<b>10</b>
11.1 MakeIndex . . . . .	10
11.2 BibT <sub>E</sub> X . . . . .	11
<b>12 Práce s obrázky</b>	<b>13</b>
12.1 Prostředí picture . . . . .	13
12.2 Krátce o METAFONTu a METAPOSTu . . . . .	13
12.3 Balíky graphics a graphicx . . . . .	14
<b>13 Jak vložit do dokumentu logo?</b>	<b>15</b>
<b>14 Tvorba diplomových a bakalářských prací</b>	<b>15</b>
<b>15 Fonty MathTime</b>	<b>15</b>
<b>16 Jak vytisknout v PostScriptu více stránek na jednu A4, tisk brožury apod.</b>	<b>15</b>







- 1 Co je to  $\text{T}_{\text{E}}\text{X}$ ?
- 2 Co je to plain $\text{T}_{\text{E}}\text{X}$ ?
- 3 O  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ u
- 4  $\text{T}_{\text{E}}\text{X}$  pod UNIXem
- 5 Editory
- 6 Kódování, čeština a spol.
- 7 Prohlížeče a konverze mezi formáty
- 8 ... a jdeme  $(\text{L}^{\text{A}})\text{T}_{\text{E}}\text{X}$ ovat. ...
- 9 Balík pro psaní českých dokumentů (czech.sty)
- 10 Program vlna
- 11 Rejstříky a citace
- 12 Práce s obrázky
- 13 Jak vložit do dokumentu logo?
- 14 Tvorba diplomových a bakalářských prací
- 15 Fonty MathTime
- 16 Jak tisknout v PostScriptu. ... ?
- 17 Literatura
- 18 Obsah

