

Návod k použití třídy cnbwp pro psaní Working Papers ČNB

Zdeněk Wagner

verze 2013.12

Abstrakt

Tento návod vysvětluje způsob psaní Working Papers České národní banky s použitím typografického systému L^AT_EX. Úvodní kapitoly se věnují popisu třídy cnbwp, jež byla pro tyto účely vytvořena. Samostatná kapitola popisuje způsob přípravy seznamu literatury. V následujících kapitolách jsou uvedeny rady pro tvorbu a vkládání obrázků a tabulek. Kapitola 9.1 obsahuje doporučení, v jaké podobě je nutné dokument odevzdat, kapitola 9.2 je vhodná pro jazykové korektory. Jména marker, balíčků, souborů a programů, o nichž se v tomto návodu píše, jsou setříděna v rejstříku. Součástí návodu jsou i vzorové soubory, jejichž výčet najdete v kapitole 10. Třída s návodem i se vzorovými soubory byly vypracovány pro Českou národní banku.

Obsah

1	Typografické konvence	2
2	Instalace	3
2.1	MikT _E X	3
2.2	teT _E X	3
2.3	T _E X Live	4
2.4	emT _E X	4
2.5	Scientific Word	4
2.6	Jiné distribuce	4
2.7	Zpracování dokumentů ze Scientific Wordu v jiných distribucích	5
3	Testování instalace	5
4	Struktura dokumentu	5
4.1	Preambule	5
4.2	Titulní stránka	7
4.2.1	Nadpis	7
4.2.2	Poděkování	8
4.2.3	Abstrakt	8
4.2.4	JEL Codes	8
4.2.5	Klíčová slova	9

4.3	Nontechnical Summary	9
4.4	Makra pro tělo dokumentu	9
5	Seznam literatury	10
5.1	Zápis pomocí BibTeXu	10
5.2	Zápis pomocí speciálních maker	10
5.2.1	Typy citovaných prací	11
5.2.2	Specifikace položek	12
5.3	Jak zvolit typ citované práce	14
5.4	Technická poznámka	15
6	Vkládání obrázků a tabulek do textu	15
6.1	Vynucené umístění plovoucího objektu na konkrétní místo	16
7	Příprava obrázků	17
7.1	Grafické formáty	17
7.2	Načtení obrázku do dokumentu	20
7.3	Příprava vektorového obrázku	21
7.3.1	Gnuplot	21
7.3.2	Corel Draw	21
7.3.3	Adobe Illustrator	21
7.4	Virtuální tiskárny	22
7.4.1	Tiskárny s výstupem do PostScriptu	22
7.4.2	Tiskárny s výstupem do PDF	22
7.5	Oprava vadných souborů EPS a PDF	22
8	Příprava tabulek	23
8.1	Zarovnání sloupce na desetinnou tečku	23
8.2	Široké tabulky	25
9	Předání dokumentu ke korektuře a ke zveřejnění	25
9.1	Příprava dokumentu k odevzdání	28
9.2	Provádění korektur	29
10	Vzorové soubory	29
11	Změny, verze 2013.12	30
A	Příloha	30
B	Index	36

1 Typografické konvence

V tomto dokumentu bude užíváno několik typů písma se speciálním významem. Text tištěný neproporcionálním písmem bude určen pro výpisy částí kódu tak, jak musí být uvedeny ve zdrojovém souboru v L^AT_EXu. Může se objevit i přímo v textu při popisu nějakého makra. V textu se tedy může vyskytnout `\usepackage` apod. Jména parametrů a jiných proměnných objektů budou zapisována *kurzívou*. V kapitole popisující způsob vytváření seznamu literatury budeme používat zápis `\typItem`, kde *typ* bude nahrazovat část jména makra. Budou tím obecně myšlena makra `\bookItem`, `\miscItem` a další.

Pro zápis jmen souborů, balíčků a tříd bude použit bezpatkový font. Příklad vidíte v názvu dokumentu i v abstraktu. Výjimkou bude zápis URL, kde potřebujeme jisté speciální znaky. URL proto budeme též tisknout neproporcionálním písmem, např.

ftp://ftp.cstug.cz/pub/tex/CTAN/, ale zde nedojde k nedorozumění. Stejným způsobem budeme zapisovat i plné názvy souborů s uvedenou cestou, např. /usr/share/texmf-dist/tex/latex/cnb/cnbwp.cls. Tyto dlouhé názvy lze rozdělit na více řádků, přičemž není uveden rozdělovník.

Zdůrazněný text bude tištěn **tučným fontem**, protože kurzíva je již použita k jinému účelu. Tučné písmo je užito i v nadpisech. V takové případě budou jména souborů, balíčků a tříd tištěna **tučným bezpatkovým písmem**.

! Odstavec, začínající velkým tučným červeným vykřičníkem, upozorňuje na velmi důležitou informaci. Nerespektování uvedeného pokynu obvykle způsobí, že při zpracování dokumentu dojde k závažné chybě, případně k chybě s exotickým hlášením, takže její příčinu nebude snadné najít.

2 Instalace

Veškerá makra pro psaní Working Papers České národní banky jsou implementována v třídě cnbwp.cls. Třída je distribuována v archivním souboru cnbwp.zip. Soubor je v archivu uložen s cestou odpovídající standardu TDS (T_EX Directory Structure). Způsob instalace se nepatrně liší v závislosti na konkrétní distribuci T_EXu.

2.1 MikT_EX

Soubor cnbwp.zip rozbalíme do adresáře X:\localtexmf, kde X: označuje disk, na němž je MikT_EX instalován (obvykle C:). Poté otevřeme menu Start/Programy/MikTeX/MikTeXOptions. Na kartě Roots si ověříme, že X:\localtexmf je v seznamu prohledávaných adresářů. Pokud není, přidáme jej. Pak stiskneme tlačítko Refresh FNDB. Úspěšnost instalace lze ověřit příkazem:

```
findtexmf cnbwp.cls
```

Při úspěšné instalaci bude vypsána plná cesta k souboru cnbwp.cls.

! Adresář X:\localtexmf je určen pro místní soubory, které nejsou standardní součástí MikT_EXu. Pokud přeinstalujete MikT_EX novou verzí, nebude tento adresář přepsán, takže o soubory nepřijdete.

2.2 teT_EX

Tato distribuce je standardem v unixových systémech a je dostupná i pro OS/2 resp. eComStation. Uvedeme si pouze instalaci v Linuxu. Instalace v OS/2 je obdobná. Liší se jen tím, že pro oddělování adresářů je užito zpětné lomítko a musíme explicitně zadat označení disku, na nějž jsme teT_EX nainstalovali.

Soubor cnbwp.zip rozbalíme do adresáře /usr/share/texmf-local. Při instalaci v unixových systémech je nutno dohlédnout na to, aby soubor /usr/share/texmf-local/tex/latex/cnb/cnbwp.cls neměl DOSové konce řádků. Po rozbalení souboru je obvykle nutné obnovit souborovou databázi příkazem mktexlsr.


Z této distribuce vychází i T_EX Live, proto vše, co bude popsáno v následující kapitole, platí i pro teT_EX.

2.3 T_EX Live

T_EX Live je oblíbenou multiplatformní distribucí T_EXu. Vychází ze stejných zdrojů jako teT_EX, instalace je tedy podobná. Adresář, do nějž rozbalíme cnbwp.zip, je většinou /usr/local/texlive/texmf-local v unixových systémech, X:\TeXLive\texmf-local ve Windows. T_EX Live však lze nainstalovat do libovolného adresáře a dokonce můžeme mít instalováno několik verzí T_EX Live současně. Adresář, do nějž máme cnbwp.zip rozbalit, zjistíme příkazem:

```
kpsewhich --expand-var=$TEXMFLOCAL
```

Adresářový strom texmf-local je sdílen všemi verzemi T_EX Live nainstalovaných na daném počítači. Při instalaci novější verze T_EX Live budou soubory nalezeny automaticky.

 Distribuce T_EX Live obvykle vyžaduje po přidání souborů spuštění programu `mktexlsr`. V distribucích pro Windows pro tento účel existuje i položka v Menu T_EX Live. Pokud programem `mktexlsr` neobnovíme databázi a před jménem adresáře s třídou cnbwp jsou uvedeny dva vykřičníky, L^AT_EX nebude schopen třídu najít. Chceme-li zjistit, zda L^AT_EX třídu najde, použijeme příkaz:

```
kpsewhich cnbwp.cls -progrname latex
```

Pokud je vše v pořádku, příkaz vypíše plnou cestu k souboru cnbwp.cls.

2.4 emT_EX

Tato distribuce je zastaralá, Eberhard Mattes ji už neudrhuje. Je vhodnější přejít na jinou distribuci. Pokud přesto chcete používat emT_EX, rozbalte cnbwp.zip do pomocného adresáře, v adresáři X:\emtex\texinput\latex2e vytvořte podadresář cnbwp a do něj zkopírujte soubory cnbwp.cls a cnbwpsizes.clo.

2.5 Scientific Word

Návod je psán pro verzi 5.5. V této verzi rozbalte cnbwp.zip do adresáře X:\sw55\TCITeX, v jiných verzích se pravděpodobně bude lišit jméno kořenového adresáře. Scientific Word nedodrhuje zcela přesně TDS, jména adresářů obsahují malá i velká písmena. Souborové systémy FAT a NTFS však malá a velká písmena nerozlišují, takže by to nemělo způsobit žádný problém.

2.6 Jiné distribuce

Při instalaci třídy cnbwp do jiných distribucí je nutno řídit se manuálem dodávaným s příslušnou distribucí. Současné distribuce jsou založeny obvykle na web2c a dodržují standard TDS, proto lze postupovat obdobně jako při instalaci pro T_EX Live, viz kap. 2.3.

2.7 Zpracování dokumentů ze Scientific Wordu v jiných distribucích

Dokumenty vytvořené Scientific Wordem vyžadují určitá makra definovaná v souborech v adresáři SWmacros. Tyto soubory jsou volně šiřitelné. Chcete-li tedy dokument zpracovat v jiné distribuci, zkopírujte celý adresář do `texmf-local/tex/latex` resp `localtexmf/tex/latex` podle toho, jakou distribuci používáte.

! Nezapomeňte, že v některých distribucích musíte po přidání souborů obnovit databázi buď příkazem `mktexlsr`, nebo z menu.

3 Testování instalace

V mnoha distribucích lze správnost instalace otestovat příkazem `kpsewhich`, jak bylo uvedeno v kapitole 2.3. Můžeme tak ověřit, že \LaTeX dokáže najít třídu `cnbwp` i makra z adresáře SWmacros. Nemáme-li program `kpsewhich` ve své distribuci, stačí napsat jednoduchý dokument:

```
\documentclass{cnbwp}
\begin{document}
Hello world.
\end{document}
```

Pokud \LaTeX při jeho zpracování nahlásí:

```
! LaTeX error: file `cnbwp.cls' not found
```

znamená to, že třída není instalována správně. V takovém případě si znovu ověřte správnost svého postupu podle příslušné kapitoly.

4 Struktura dokumentu

Working paper je běžným dokumentem psaným v \LaTeX u, skládá se tedy z obvyklých částí: z preambule a z těla dokumentu. Do preambule patří vše, co je uvedeno před příkazem `\begin{document}`. Zvláštní částí těla dokumentu je seznam literatury. Tomu bude věnována samostatná kapitola 5.

4.1 Preambule

Všechna základní makra jsou implementována v třídě `cnbwp`. Každý dokument musí proto začínat příkazem:

```
\documentclass{cnbwp}
```

Příkaz `\documentclass` umožňuje zadávání parametrů v hranatých závorkách před jménem třídy. Tak je tomu i v tomto případě. Třída `cnbwp` je odvozena z třídy `article` a přebírá všechny parametry. Formát Working Papers je však pevně stanoven, proto parametry, jež mění rozměry papíru, velikost písma a počet sloupců, jsou ignorovány.

V třídě `cnbwp` lze použít několik nových parametrů. První z nich se týká číslování obrázků a tabulek. Standardně jsou tabulky a obrázky číslovány nehierarchicky. Způsob číslování lze volit. Nastavuje se těmito parametry:

hierarchicalnumbering hierarchické číslování
simplenumbering jednoduché číslování

Pokud není žádný z těchto parametrů uveden, případně jsou uvedeny oba, bude se používat jednoduché číslování.

Popisky obrázků a tabulek jsou standardně zarovnány na levý okraj. V některých dokumentech může vypadat esteticky lépe, budou-li popisy zarovnány na střed. Zarovnání pro celý dokument nastavíme těmito parametry:

standardcaptions zarovnání popisků vlevo
centeredcaptions zarovnání popisků na střed

Není-li uveden žádný z těchto parametrů, nebo jsou uvedeny oba, budou popisky obrázků a tabulek zarovnány vlevo.

Za určitých okolností je nutné některé popisky zarovnat jinak než ostatní popisy v dokumentu. Pro tento účel slouží makra, která budou vysvětlena později v kapitole 4.4.

Třída `cnbwp` načítá implicitně balíčky, které jsou pro formátování Working Papers nezbytné. Jsou to `ifpdf`, `mathptmx`, `fontenc` s parametrem `T1`, `babel` s parametry `czech` a `english`, `natbib`, `url` a `keyval`. Parametry, uvedené v hranatých závorkách v příkazu `\documentclass`, jsou automaticky poslány všem těmto balíčkům, jakož i dalším balíčkům, které si zavedete sami později. Nejčastěji budete pravděpodobně specifikovat parametry pro balíček `natbib`. Budete-li chtít změnit zápis citací z formátu autor-rok na číslované, použijete:

```
\documentclass [numbered] {cnbwp}
```

Nyní můžete příkazem `\usepackage` načíst další balíčky potřebné pro formátování svého dokumentu. Lze uvést v jedné příkazu `\usepackage` více balíčků oddělených čárkami, např.:

```
\usepackage{graphicx,amsmath,dcolumn}
```

Makro `\usepackage` též umožňuje uvádění parametrů v hranatých závorkách. Tyto parametry jsou poslány pouze balíčkům uvedeným v daném příkaze. Pak ovšem nelze stejným příkazem načíst více balíčků, protože by \LaTeX hlásil, že parametry nejsou v balíčku implementovány. Musíme použít např.:

```
\usepackage{amsmath,dcolumn}  
\usepackage[dvips]{graphicx}  
\usepackage[figuresright]{rotating}
```

Všem balíčkům budou poslány též parametry uvedené v příkazu `\documentclass`. Parametr `draft` lze tedy specifikovat globálně:

```
\documentclass[draft]{cnbwp}
\usepackage{graphicx}
```

Můžeme jej však také použít jen s balíčkem `graphicx`:

```
\documentclass{cnbwp}
\usepackage[draft]{graphicx}
```

V obou případech budou obrázky nahrazeny obdélníkovým rámečkem velikosti obrázku, v němž bude vytištěno jméno souboru. V prvním případě budou navíc přetečené (overful) boxy označeny na okraji černými obdélníčky.

! Balíček `hyperref` používá jinou definici makra `\url` než balíček `url`. Tato definice navíc způsobuje konflikty při zápisu odkazů. Třída `cnbwp` proto zkopíruje definici `\url` z balíčku `url` do pomocného makra `\CNBurl` a po provedení `\begin{document}` je definice makra `\url` obnovena tak, aby odpovídala balíčku `url`. Chcete-li používat makro z balíčku `hyperref`, uložte si jej po zavedení balíčku do makra s vhodným názvem, např. pomocí:

```
\let\MYurl\url
```

! Nikdy neměňte definici makra `\url`. Makra pro formátování seznamu literatury na jeho definici spoléhají. Makro `\url` může být použito v databázi citací zpracovávané `BibTeXem`, takže v dokumentu jej nemusíte přímo vidět. Při změně definice bude `LATEX` hlásit podivné chyby a formátování části dokumentu se pravděpodobně zcela zhroutí.

! `LATEX` nevyžaduje používání oddělených jmenných souborů pro jednotlivé balíčky. Může se tedy stát, že balíčky obsahují konfliktní definice, jež způsobí těžko předvídatelné chyby. Kromě výše zmíněné kolize nelze současně s balíčkem `url` použít balíček `path`. Proto je vhodné začínat s minimálním dokumentem a používat jen ty balíčky, jež jsou pro zpracování skutečně nezbytné. Kolize je nutno hledat a odstraňovat metodou pokusu a omylu. Někdy pomůže i změna pořadí zavádění balíčků.

V závěru preambule je vhodné definovat vlastní makra potřebná v celém dokumentu. Pro definici maker se zvlášť hodí příkaz `\DeclareRobustCommand`. Má stejnou syntaxi jako `\newcommand`. Liší se jen tím, že definované makro je robustní. Při použití takto definovaného makra v příkazech `\section` a `\subsection` nemusíte používat `\protect`.

4.2 Titulní stránka

Working Paper musí vždy začínat titulní stránkou, která se skládá z několika součástí. Jim budou věnovány následující podkapitoly.

4.2.1 Nadpis

Nadpis je na pomezí preambule a těla dokumentu. Ve skutečnosti některá makra lze uvést již v preambuli. Přesněji řečeno, makra, definující název, jména autorů a poděkování, lze zapsat kamkoliv mezi `\documentclass` a `\maketitle`. Jejich uvedení na samém začátku souboru tedy může být užitečné.

Název práce zadáváme jako argument makra `\title`. Název bude vytištěn později, až uvedeme příkaz `\maketitle`, ale dostane se též do záhlaví:

```
\title{Fidlovačka aneb žádný hněv a žádná rvačka}
```

Jména autorů zapisujeme pomocí makra `\author`, které vyžaduje dva parametry. V prvním parametru je uvedeno plné jméno, v druhém parametru název instituce. Makro se uvede pro každého autora samostatně např. takto:

```
\author{Kapitán Nemo}{Nautilus}  
\author{Robinson Crusoe}{Pustý ostrov}
```

Makro `\maketitle` doplní před jméno posledního autora spojku „and“ podle britských pravidel, tj. čárka je uvedena pouze v případě, že práce má více než dva autory. Jména autorů jsou též přenesena do záhlaví.

4.2.2 Poděkování

Poděkování je nepovinnou součástí Working Paper. Uvádíme jej v argumentu makra `\acknowledge`.

Název práce bude vytištěn uvedením makra `\maketitle`. Toto makro se musí vyskytovat až za `\begin{document}`.

4.2.3 Abstrakt

Práce musí obsahovat abstrakt ve dvou jazycích, nejprve anglicky, pak česky. Anglický abstrakt uvádíme v prostředí `abstract`, český v prostředí `abstrakt`, tj. například takto:

```
\begin{abstract}  
Place the abstract here.
```

```
The abstract may contain more than one paragraph. A blank line should be left  
between the paragraphs. No blank line is needed at the end of the abstract.  
\end{abstract}
```

```
\begin{abstrakt}  
Zde uvedeme abstrakt práce.
```

```
Abstrakt může obsahovat více odstavců. Mezi odstavci ponecháme  
prázdný řádek. Na konci abstraktu prázdný řádek být nemusí.  
\end{abstrakt}
```

V prostředí `abstrakt` je automaticky aktivováno české dělení slov.

4.2.4 JEL Codes

JEL codes uvádíme v makru `\JEL`:

```
\JEL{E22, E23, E32, E52}
```


4.2.5 Klíčová slova

Klíčová slova jsou posledním elementem titulní stránky. Zadáváme je pomocí výše zmíněného makra `\Keywords`:

```
\Keywords{Nautilus, moře, 2000 mil, velryba, malström}
```

4.3 Nontechnical Summary

Nontechnical Summary je text v rozsahu přibližně jedné stránky a v dokumentu je vytištěn na samostatné straně. Zapisuje se, podobně jako abstrakt, do prostředí `nontechsummary`:

```
\begin{nontechsummary}
```

V tomto prostředí zapisujeme Nontechnical Summary. Nadpis bude vyvořen automaticky a nebude číslován.

Text se obvykle skládá z několika odstavců, jež jsou od sebe odděleny prázdným řádkem.

```
\end{nontechsummary}
```

! Makro `\Keywords` a prostředí `nontechsummary` vynucují přechod na novou stránku. Pokud oba tyto elementy chybí a stránkový zlom je vynucen ručně příkazem `\newpage` nebo `\clearpage`, nebude část dokumentu naformátována správně.

4.4 Makra pro tělo dokumentu

Tělo dokumentu je psáno stejně jako by byla použita běžná třída `article`. Sekce tedy uvádíme makrem `\section`, podsekce `\subsection`. Je implementován i příkaz `\subsubsection`. S členěním až na `\paragraph` a `\subparagraph` nebylo při návrhu počítáno, ale syntaktická chyba při použití těchto maker nevznikne.

Popisky tabulek a obrázků zapisujeme do makra `\caption`. Popisek bude zarovnán podle toho, jaký parametr jsme uvedli v preambuli (viz str. 6). Chceme-li jeden popisek zarovnat jinak, uvedeme explicitně `\standardcaption` pro zarovnání vlevo a `\centeredcaption` pro zarovnání na střed. Způsob číslování vždy odpovídá parametrům uvedeným v preambuli, jež jsou vysvětlena na straně 6. V dokumentu jej změnit nelze. Z typografického hlediska to ani nemá smysl.

V popisu tabulky či obrázku lze uvést poznámku pomocí makra `\Note` a informaci o zdroji pomocí makra `\Source`. Tato makra se zpravidla uvádějí až pod makrem `\caption` podle tohoto příkladu:

```
\Note{Value Added Tax (VAT), Excises (E), Personal Income Tax (PIT),  
      Social Security Contributions (SSC), Inheritance Tax (IT),  
      Corporate Income Tax (CIT), Other Age-Specific Revenues (OR).}  
\Source{CNB}
```

5 Seznam literatury

Seznam literatury (bibliografii) lze zapisovat dvěma způsoby. V obou případech je k vlastnímu formátování využít balíček `natbib`, jenž je automaticky načítán třídou `cnbwp`. Vzhled dokumentu tedy nezávisí na tom, která metoda byla pro zápis seznamu literatury zvolena.

V kapitole 5.1 bude stručně uveden způsob vytváření seznamu literatury při použití `BibTeXu`. V kapitole 5.2 a jejích podkapitolách budou podrobně vysvětlena makra vytvořená pro `Working Papers České národní banky`.

5.1 Zápis pomocí `BibTeXu`

Použití `BibTeXu` je popsáno v dokumentu Orena Patashnika „`BibTeXing`“, jenž je součástí běžných distribucí `TeXu`. Využijeme stylu `abbrvcnb` odvozeného z bibliografických stylů balíčku `natbib`. Předpokládejme, že bibliografickou databázi máme uloženu v souboru `odkazy.bib`. V `LATEX`ovém dokumentu pak uvedeme:

```
\bibliographystyle{abbrvcnb}
\bibliography{odkazy}
```

Po zpracování dokumentu `LATEX`em spustíme `BibTeX`:

```
bibtex dokument
```

(Místo slova *dokument* doplňte název souboru, zde jsme předpokládali, že dokument je uložen v souboru `dokument.tex`.) Pak je nutno zpracovat dokument dvakrát `LATEX`em. Výjimečně je nutný ještě třetí průchod, pokud po doplnění odkazů do textu došlo ke stránkovému posunu.

URL dokumentů jsou většinou dlouhé řetězce, v nichž nefunguje automatické dělení slov, případně se slova dělí na nevhodných místech a může dojít ke zmatení. Problém je řešen v souladu se standardy pro zápis URL v balíčku `url`, který je též implicitně načítán třídou `cnbwp`. URL se obvykle vyskytuje v položce *note*. Tato položka by tedy měla být psána způsobem:

```
note = "\url{ftp://ftp.cstug.cz/pub/tex/CTAN/}",
```

5.2 Zápis pomocí speciálních maker

Seznam literatury můžeme psát speciálními makry přímo v `LATEX`ovém souboru v prostředí `thebibliography`. Toto prostředí má jeden povinný parametr, z něž se určuje šířka potřebná pro pořadové číslo. `BibTeX` do tohoto parametru zapíše počet citací. Lze však uvést jakýkoliv text vhodné šířky, například `XX`, jak je použito v příkladu (viz Příloha). Jednotlivé citace jsou zapisovány ve tvaru:

```
\typItem[text citace]{návěští}{obsah citace}
```

Místo slova *typ* musíme uvést odpovídající typ citované práce. Použijeme identifikátor odvozený ze standardních typů definovaných v `BibTeXu`. Jejich seznam a popis bude uveden v kapitole 5.2.1.

První dva parametry (nepovinný a povinný) mají stejný význam jako u makra `\bibitem`. Povinný parametr *návěští* je využíván při citování v textu v makru `\cite`, nepovinný parametr *text citace* obsahuje údaj, který je v místě citace zapsán v textu. Tento parametr má obecně tvar:

[*krátký text (rok) dlouhý text*]

Balíček `natbib` dokáže volit mezi uvedením krátkého a dlouhého textu. Pokud má citovaná práce pouze jednoho autora, *dlouhý text* se neuvádí. Celý parametr pak můžeme zapsat ve tvaru:

[Pytlík(1984)]

Obě varianty se uvádějí u prací s více autory. Chceme-li například citovat knihu vydanou v roce 1984, jejímiž autory jsou Brouk Pytlík, Ferda Mravenec a Bába Krtonožka, zapíšeme parametr ve tvaru:

[Pytlík et.~al.(1984)Pytlík, Mravenec a Krtonožka]

! Při zpracování Bib \TeX em jsou informace do makra `\typItem` vytvářeny automaticky. Pokud zadáváme informace ručně, musíme správně doplnit parametr v hranatých závorkách. Informace v kulatých závorkách tedy musí být identická s hodnotou pole `year`, jak je uvedeno mimo jiné v prvních dvou položkách v příkladu v příloze A. Bib \TeX seřadí citace abecedně, při přímém zápisu musí citace správně seřadit autor.

Při volbě formátu textu citace vycházíme z požadavků, jak má výsledný text vypadat. Pokud nepožadujeme dlouhou variantu odkazů, nemusíme ji v makru `\typItem` uvádět.

Balíček `natbib` nutně vyžaduje kulaté závorky v nepovinném parametru. Pokud není znám rok zveřejnění citované práce, musíme uvést alespoň prázdné závorky `()`.

V parametru *obsah citace* zadáváme odkaz na citovanou práci. Formát parametru bude popsán v kapitole 5.2.2.

5.2.1 Typy citovaných prací

Makra použitá pro citace prací vycházejí ze standardních typů definovaných v Bib \TeX u. Zde uvedeme jejich výčet a stručný popis včetně seznamu povinných a nepovinných položek. Význam položek a způsob jejich zápisu bude vysvětlen v následující kapitole.

`\articleItem` článek v časopisu. Povinné položky: `author`, `title`, `journal`, `year`. Nepovinné položky: `volume`, `number`, `pages`, `month`, `note`.

`\bookItem` kniha s uvedeným vydavatelem. Povinné položky: `author` nebo `editor`, `title`, `publisher`, `year`. Nepovinné položky: `volume` nebo `number`, `series`, `address`, `edition`, `month`, `note`.

`\bookletItem` vytištěná a svázaná publikace bez uvedeného nakladatele a instituce. Povinná položka: `title`. Nepovinné položky: `author`, `howpublished`, `address`, `edition`, `month`, `year`, `note`.

`\conferenceItem` viz `\inproceedingsItem`.

`\inbookItem` část knihy, obvykle kapitola nebo rozmezí stránek. Povinné položky: `author` nebo `editor`, `title`, `chapter` a/nebo `pages`, `publisher`, `year`. Nepovinné položky: `volume` nebo `number`, `series`, `type`, `address`, `edition`, `month`, `note`. Význam položky `type` bude vysvětlen na straně 14.

`\incollectionItem` část knihy mající vlastní název. Povinné položky: `author`, `title`, `booktitle`, `publisher`, `year`. Nepovinné položky: `editor`, `volume` nebo `number`, `series`, `type`, `chapter`, `pages`, `address`, `edition`, `month`, `note`. Položka `type` má stejný význam jako u předchozího typu citace a je vysvětlena na straně 14.

`\inproceedingsItem` článek ve sborníku konference. Povinné položky: `author`, `title`, `booktitle`, `year`. Nepovinné položky: `editor`, `volume` nebo `number`, `series`, `pages`, `address`, `edition`, `month`, `organization`, `publisher`, `note`.

`\manualItem` technická dokumentace. Povinná položka: `title`. Nepovinné položky: `author`, `organization`, `address`, `edition`, `month`, `year`, `note`.

`\mastersthesisItem` diplomová práce. Povinné položky: `author`, `title`, `school`, `year`. Nepovinné položky: `type`, `address`, `month`, `note`.

`\miscItem` jiný typ užívaný v případě, že se nic jiného nehodí. Má pouze nepovinné položky `author`, `title`, `howpublished`, `month`, `year`, `note`.

`\phdthesisItem` disertační práce. Povinné položky: `author`, `title`, `school`, `year`. Nepovinné položky: `type`, `address`, `month`, `note`.

`\proceedingsItem` sborník konference. Povinné položky: `title`, `year`. Nepovinné položky: `editor`, `volume` nebo `number`, `series`, `address`, `month`, `organization`, `publisher`, `note`.

`\techreportItem` zpráva publikovaná školou nebo jinou institucí obvykle v číslované sérii. Povinné položky: `author`, `title`, `institution`, `year`. Nepovinné položky: `type`, `number`, `address`, `month`, `note`.

`\unpublishedItem` dokument mající autora a název, avšak formálně nepublikovaný. Povinné položky: `author`, `title`, `note`. Nepovinné položky: `month`, `year`.

5.2.2 Specifikace položek

V posledním parametru maker `\typItem` je zapsán obsah jednotlivých položek citované práce. Položky se uvádějí ve tvaru:

```
{jméno1 = hodnota1, jméno2 = hodnota2, jméno3 = hodnota3}
```

Mezery okolo znaků rovnítek a čárek, oddělujících jednotlivé položky, jsou ignorovány. Ignorují se též prázdné položky. Lze tedy napsat více čárek za sebou, čárku před první položkou a za poslední položkou. Všude, kde je povolena mezera, smíme též ukončit řádek. Kromě toho \TeX ignoruje všechny mezery na počátku řádku. Seznam literatury proto můžeme zapisovat přehledně podobným stylem, jaký je použit v příloze.

Položky se vzájemně oddělují čárkami. Pokud má být uvedena čárka v hodnotě položky, musíme celou hodnotu uzavřít do složených závorek. Tento případ se bude nejčastěji vyskytovat při citaci prací s více autory. Má-li publikace pouze dva autory, jejichž jména jsou spojena spojkou *a* (v anglických pracích spojkou *and*), budeme psát

author = Ferda Mravenec a Brouk Pytlík, ...

Zde čárka ukončuje položku. Máme-li více autorů, musíme jejich jména uzavřít do závorek takto:

author = {Ferda Mravenec, Brouk Pytlík a Bába Krtonožka}, ...

První čárka (uvnitř závorek) je součástí seznamu autorů, druhá čárka ukončuje položku. V anglických pracích uvádíme čárku před *and* v případě, kdy citace má více než dva autory.

! Zapomenutá čárka nebo vynechané závorky okolo hodnoty obsahující čárku mají za následek chybové hlášení o neznámé položce s podivným jménem. Někdy se kvůli chybám tohoto typu může dokonce ztratit velká část seznamu literatury i následujících částí dokumentu.

V databázových souborech určených pro zpracování Bib_TE_Xem často používáme složené závorky, jimiž zabraňujeme konverzi na malá či velká písmena. L^AT_EX žádné konverze neprovádí, proto závorky nepotřebujeme. Můžeme však použít různá makra, např. `\mbox` pro zabránění nevhodnému dělení, `\-` k určení vhodného místa dělení slova, a dokonce si můžeme přímo uvnitř prostředí `thebibliography` definovat vlastní makra, jež budou v ostatních částech dokumentu neznámá (nebo definovaná jinak).

Položky mohou být zapsány v libovolném pořadí, lze však použít pouze následující:

address adresa nakladatele nebo instituce (např. školy u diplomových a disertačních prací)

author jména autorů v tom formátu, jak mají být uvedena, na rozdíl od Bib_TE_Xu nedochází k žádnému přeformátování (křestní jméno nebude zkráceno na iniciálu apod.).

booktitle název knihy, jejíž část je citována.

chapter číslo kapitoly.

edition vydání, mělo by to být pořadové číslo, které může být zadáno i slovně, např. *Second*.

editor jména editorů, zapisují se ve stejném formátu jako jména autorů.

howpublished způsob, jak byla zveřejněna práce neobvyklého typu.

institution instituce, která sponzorovala technickou zprávu (`\techreportItem`).

journal jméno časopisu, resp. jeho zkratka.

month měsíc zveřejnění práce, v této položce lze uvést i den.

note vysvětlující poznámka. Tato položka je sice uvedena v příkladech v příloze, ale v praxi ji budete užívat jen zřídka. Slouží k uvedení URL prací zveřejněných na Internetu, přičemž pro automatické a správné dělení dlouhých URL na více řádků se používá balíček `url` načítaný třídou `cnbwp`. URL tedy budete zadávat ve tvaru:

```
note = \url{ftp://ftp.cstug.cz/pub/tex/CTAN/}, ...
```

number číslo časopisu, technické zprávy nebo práce vydávané v sérii.

organization organizace, která pořádala konferenci, nebo vydala manuál.

pages rozmezí stran.

publisher jméno vydavatele.

school škola, kde byla obhájena diplomová nebo disertační práce.

series jméno série (edice) knih.

title název práce.

type typ práce, např. *Research Note*. V citacích `\inbookItem` a `\incollectionItem` specifikuje text odpovídající položce `chapter`. Standardně se doplní `chapter`, ale můžete zadat např. `section` nebo `kapitola`.

volume svazek časopisu nebo vícesvazkové knihy.

year rok vydání, u nepublikovaných prací rok napsání.

Členění na položky nemusí být dodržováno zcela striktně. Můžeme informace oddělit do samostatných položek:

```
publisher = John Wiley \& Sons, address = New York, ...
```

Seznam literatury, zapsaný přímo v \LaTeX ovém dokumentu, zřejmě nebude sloužit pro další strojové zpracování. Může být proto účelné spojení obou informací do jedné položky:

```
publisher = {John Wiley \& Sons, New York}, ...
```

Zde jsme museli použít závorky, protože spojená informace obsahuje čárku.

Makra automaticky doplňují tečky nebo čárky za jednotlivé položky, ale pouze v případě, že text nekončí interpunkčním znaménkem. Mějme název zapsaný:

```
title = Can we spend less money for more music?, ...
```

V tomto případě nebude interpunkční znaménko doplněno, a to ani v případě, kdyby celý název byl uzavřen ve složených závorkách.

5.3 Jak zvolit typ citované práce

Typ citované práce musíme zvolit jak při zápisu do databáze pro $\text{Bib}\TeX$, tak při použití maker `\typItem`. Názvy typů jsou intuitivní a správná volba by měla být snadná. Ve sporných případech pomůže, když si uvědomíme, které položky jsou povinné a které nepovinné. Zvolíme takový typ, jehož struktura odpovídá účelu.

Některé položky jsou zdánlivě nadbytečné, např. diplomové (`\mastersthesisItem`) a disertační (`\phdthesisItem`)¹ práce jsou formátovány zcela stejně, liší se jen defaultní hodnota nepovinné položky `type`. Chceme-li citovat obdobnou publikaci, např. habilitační práci, můžeme použít kterýkoliv typ a specifikujeme nepovinnou položku:

```
type = Habilitační práce, ...
```

Práce dostupné pouze z Internetu nejsou formálně publikovány. Používá se pro ně `\unpublishedItem`, v $\text{Bib}\TeX$ u `@unpublished`. URL práce se uvede v položce `note`, jak bylo uvedeno výše.

¹V $\text{Bib}\TeX$ u `@mastersthesis` a `@phdthesis`.

5.4 Technická poznámka

V příkazu `\documentclass` lze uvádět nepovinné parametry v hranatých závorkách. Tyto parametry jsou posílány jak třídě `article`, z níž je třída `cnbwp` odvozena, tak všem načítaným balíčkům. Lze tedy specifikovat parametry pro balíček `natbib`. Například parametrem `numbered` změníme citace ve tvaru autor-rok na číslované.

! Poslední parametr maker `\typItem` je zpracován pomocí balíčku `keyval`. Z toho vyplývá, že smíme používat výhradně deklarované položky. Nelze vytvořit neznámou položku pro přidání komentáře, který se nemá tisknout. Takový pokus vede k chybě při zpracování dokumentu. Komentáře zapisujeme za znak *procento* jako v běžných L^AT_EXových textech.

Nepovinný parametr a první povinný parametr maker `\typItem` jsou beze změny předány makru `\bibitem`.

V případě, že požadovanému účelu nevyhovuje žádný z typů uvedených v kapitole 5.2.1, lze použít přímo makro `\bibitem` a vše naformátovat ručně.

6 Vkládání obrázků a tabulek do textu

Obrázky a tabulky vkládáme do textu prostřednictvím plovoucích prostředí `figure` a `table`. V hranatých závorkách uvádíme nepovinně parametr, kde má být plovoucí objekt umístěn. Tyto parametry L^AT_EX ovšem chápe pouze jako doporučení. Nelze-li jim vyhovět, pokusí se o umístění plovoucího objektu na samostatnou stránku. Není-li ani to možné, umístí příslušný plovoucí objekt a všechny následující na konec dokumentu resp. kapitoly (v případě knih). Je tedy nutno povolit více možností, aby si L^AT_EX mohl vybrat. Uvedení samotného specifikátoru `[h]` obvykle vede právě k tomu, že je plovoucí objekt odsunut na samostatnou stránku či konec dokumentu. O vynucené umístění plovoucího objektu na konkrétní místo si povíme v kapitole 6.1.

! Zde je vhodné místo k vysvětlení rozdílu mezi makry `\newpage` a `\clearpage` či `\cleardoublepage`. První dvě vynutí přechod na novou stránku, `\cleardoublepage` na lichou stránku². Makra `\clearpage` a `\cleardoublepage` navíc vytisknou všechny dosud neumístěné plovoucí objekty uschované ve vyrovnávací paměti.

Algoritmus umísťování plovoucích objektů se řídí hodnotami několika registrů. Hodnoty jsou voleny pro běžné dokumenty, ale pro technické zprávy s mnoha tabulkami a obrázky nemusí být vyhovující. Registr `\topfraction` určuje maximální část stránky, kterou smí zabrat plovoucí objekt nahoře, `\bottomfraction` analogicky vymezuje maximální podíl stránky pro plovoucí objekty umístěné dole a `\textfraction` je minimální část stránky, již musí zabírat text. Registry se mění příkazem `\renewcommand`, neboť se jedná o makra. V dokumentech s mnoha plovoucími objekty je můžeme nastavit na extrémní hodnoty pomocí:

```
\renewcommand\topfraction{.99}
\renewcommand\bottomfraction{.99}
\renewcommand\textfraction{.01}
```

²To platí pouze pro oboustranně tištěné dokumenty, které mají explicitně či implicitně specifikován parametr `twoside`. Třída `cnbwp` jej deklaruje implicitně.

Plovoucí stránka je vytvořena pouze tehdy, pokud plovoucí objekty zaujímají alespoň část definovanou v registru `\floatpagefraction`. Tuto hodnotu obvykle není nutno měnit.

L^AT_EX má ještě čítače, jimiž nastavujeme nejvyšší povolený počet plovoucích objektů. Čítač `topnumber` určuje maximální počet plovoucích objektů nahoře, `bottomnumber` dole a `\totalnumber` omezuje celkový počet všech plovoucích objektů na stránce. Obvykle je nemusíte měnit, k modifikacím by bylo nuno přistoupit jen v případě, že by dokument obsahoval velký počet velmi malých plovoucích objektů³. Hodnoty se nastavují příkazem `\setcounter`, například:

```
\setcounter{totalnumber}{17}
```

! Aktuální hodnoty registrů lze zjistit různými způsoby. Zde si předvedeme použití příkazů T_EXu `\show` a `\showthe`. První z nich slouží k zobrazení definice makra. Použijeme jej tedy ke zjištění hodnoty registru typu *fraction*. Registr typu *number* musíme vypsat pomocí `\showthe`. Následující dva řádky vypíší aktuální hodnoty registrů `\topfraction` a `\topnumber`:

```
\show\topfraction
\showthe\value{topnumber}
```

Na terminálu a v log-souboru (zde jsme vypustili prázdné řádky) pak najdeme výpis podobný tomuto:

```
> \topfraction=\long macro:
->.7.
1.875 \show\topfraction

> 2.
<recently read> \c@topnumber
1.876 \showthe\value{topnumber}
```

Zjistili jsme tedy, že `\topfraction = 0.7` a `topnumber = 2`. Tečky na konec definic přidávají příkazy `\show` a `\showthe`.

6.1 Vynucené umístění plovoucího objektu na konkrétní místo

V některých případech se stává, že plovoucí objekt **musí** nezbytně být na jistém konkrétním místě. Potřebujeme tedy L^AT_EXu sdělit, že specifikátor nemyslíme jako doporučený, nýbrž jako kategorický příkaz. K tomu slouží specifikátor [H]. Původně byl implementován v balíčku `here`, ale ten již není v běžných distribucích. Tento specifikátor je nyní implementován v balíčku `float`.

! Při použití specifikátoru [H] prostředí přestává být plovoucím. L^AT_EX jej umístí na požadované místo bez ohledu na to, zda se do zbývajících prostoru na stránce vejde. Používejte proto tento specifikátor až ve finální verzi dokumentu.

³Pak by ale stálo za úvahu jejich seskupení s využitím balíčku `subfigure`.

7 Příprava obrázků

Tato kapitola se bude věnovat práci s obrázky. Nejprve si vysvětlíme rozdíly mezi hlavními grafickými formáty. Dále si popíšeme, jak načteme obrázky do dokumentu, a na konec uvedeme několik postupů, jak obrázky připravovat a jak opravit obrázky vytvořené špatně napsanými programy.

7.1 Grafické formáty

Obrázky podle formátu dělíme na dva základní typy: bitmapové a vektorové. Bitmapové obrázky představují matici hodnot (rastr), přičemž každý prvek nese informaci o barvě zobrazeného bodu nenulové velikosti. Vektorové obrázky se liší tím, že jsou v nich křivky popsány matematickými rovnicemi a ke každé křivce je přidána informace o tloušťce čáry, její barvě, případně o barvě, již má být vyplněna ohraničená plocha. K převodu na rastr dochází až v okamžiku zobrazení na monitoru či tisku.

Zdálo by se, že není velký rozdíl ve výsledku, protože vektorový obrázek se také musí vyrašťovat, ale opak je pravdou. Zásadní rozdíl je v tom, že každé zobrazovací či tiskové zařízení má jiné rozlišení, tj. jinou hustotu rastru. Vektorový obrázek se rastruje v okamžiku zobrazení, je tedy rastrován na konkrétní rozlišení. Záleží též na velikosti a tvaru tiskových bodů. Podle toho se určuje barva bodů, jež jsou matematicky na hranici mezi obarvenou a neobarvenou částí roviny. Rasterizační algoritmus zobrazovacího zařízení je vypočte tak, aby křivky vypadaly hladce.

Při zobrazení bitmapového obrázku je nutno zachovat předepsanou velikost, přestože byl vytvořen pro jiné rozlišení. Nezbyvá tedy jiná možnost než body dopočítat interpolací. V bitmapovém obrázku již nemáme informaci, zda bod ležel na nějaké křivce. Přestože ovladač příslušného zařízení zná velikost a tvar tiskových bodů, nemá dost informací, aby je při rastrování mohl použít.

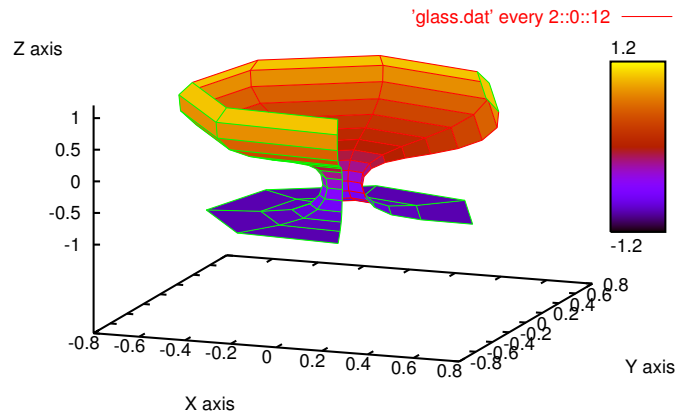
Příklad vektorového a bitmapového grafu je uveden na obrázku 1. Obrázek je jedním z demonstračních příkladů dodávaných s programem gnuplot. Ze stejného zdroje byla vytvořena vektorová i bitmapová podoba.

Horní obrázek je vektorový. Byl exportován jako EPS (Encapsulated PostScript) a převeden na PDF, aby mohl být načten pdf \TeX em nebo \TeX em. Na obrazovce mohou být křivky zubaté, protože rozlišení obrazovky je malé, ale po vytištění na kvalitní tiskárně zuby zmizí.

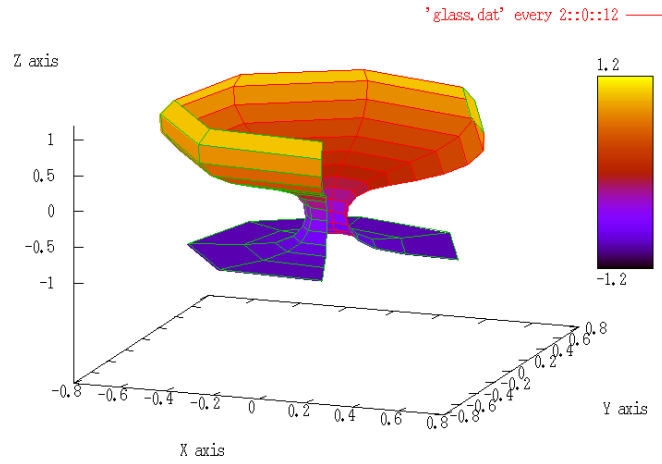
Střední obrázek byl exportován ve formátu PNG (Portable Network Graphics). Rozdíl v kvalitě zobrazení je zřejmý, zejména výrazné je zhoršení kvality písma. Při určitém zvětšení mohou zcela zmizet osy. Osy zůstanou zubaté i po vytištění na tiskárně s vysokým rozlišením, neboť, jak již bylo napsáno, ovladač příslušného zařízení neví, že body měly ležet na jedné přímce, a pouze „schody“ přepočítá na jiné rozlišení.

Rastrová data zabírají příliš mnoho paměti. V obrázcích se však vyskytují opakované sekvence barev a jednobarevné plochy. Data lze tedy uložit ekonomičtěji využitím nějakého kompresního algoritmu. Kompresní algoritmy dělíme do dvou skupin, na bezztrátové a ztrátové. Bezztrátové algoritmy se vyznačují tím, že lze dekompresí obnovit původní data. U ztrátových algoritmů to v principu nelze.

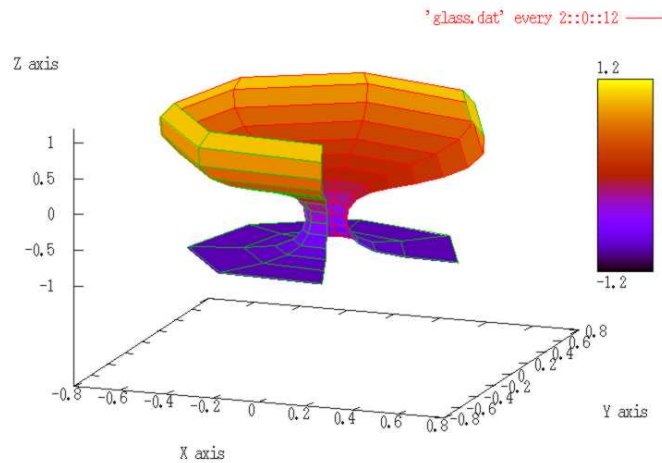
splot with "set pm3d" (implemented with some terminals)



splot with "set pm3d" (implemented with some terminals)



splot with "set pm3d" (implemented with some terminals)



Obrázek 1: Srovnání vektorového obrázku (nahore) s bitmapovým s bezztrátovou kompresí (uprostřed) a ztrátovou kompresí JPEG (dole)

Formát PNG, použitý na středním obrázku, používá bezztrátový kompresní algoritmus. Spodní obrázek byl uložen ve formátu využívajícím ztrátovou kompresi JPEG⁴. Tento algoritmus vyvinulo konsorcium Joint Photographic Expert Group pro ukládání barevných fotografií. Výzkumem bylo zjištěno, které obrazové elementy lidské oko nerozezná. Není tedy na závadu, jsou-li z fotografie odstraněny. Kvalita není zhoršena a výrazně se zmenší velikost souboru. Kvalitu obrázku lze ovlivnit, ale i při nejvyšší kvalitě zobrazení (nejméně účinné kompresi) je vždy komprese ztrátová. Formát se nazývá JFIF (JPEG File Interchange Format) a soubory mají nejčastěji příponu jpg, méně často jpeg, jff nebo jfif.

Spodní obrázek byl vytvořen v nízkém rozlišení, vhodném pouze pro obrazovku, a uložen úmyslně s nastavenou nízkou kvalitou, aby bylo na první pohled zřejmé, že při použití ztrátové komprese dochází k rozmazání a k vytvoření barevných artefaktů, zejména v téměř jednobarevných plochách.

! Pamatujme si, že obrázky vždy vytváříme v takovém formátu, který nejlépe odpovídá jejich charakteru. Grafy a diagramy jsou vždy matematické objekty, které lze popsat rovnicemi známými z analytické geometrie. Komplikované křivky jsou přitom obvykle aproximovány Bézierovými křivkami. Obrázky tedy vytvoříme ve vektorovém formátu. Bitmapové formáty používáme pouze pro obrázky, které nemají matematickou reprezentaci. Přednost dáváme formátům s bezztrátovou kompresí. Grafické formáty se ztrátovou kompresí používáme **výhradně na barevné fotografie a na nic jiného!**

Ohledně grafických formátů panuje mnoho nepravdivých pověr. Jednou z nich je domněnka, že formát EPS je vektorový. Je to pravda pouze částečná. Formát EPS je v zásadě PostScript s jistými omezeními. Soubor může obsahovat jak vektorovou, tak bitmapovou grafiku, a to dokonce jejich kombinaci. Při ukládání obrázku ve formátu EPS tedy záleží na tom, jakým programem jsme jej vytvořili. Pokud byl obrázek nakreslen bitmapovým editorem, jakými jsou např. Photoshop nebo Gimp, bude i výsledný EPS bitmapový. Totéž platí o formátu PDF. Oba formáty povolují u bitmapových obrázků bezztrátovou i ztrátovou kompresi.

Mnoho lidí se omylem domnívá, že není rozdíl mezi formáty PS (PostScript) a EPS. Bohužel se tímto velmi oblíbeným omylem nechali svést i někteří autoři komerčních programů. EPS povoluje pouze podmnožinu jazyka PostScript a vyžaduje nějaké informace navíc, aby programy dokázaly vložit obrázek do dokumentu. Některé programy však vytvářejí postscriptové soubory, jež se jen tváří jako EPS. V některých případech to nevadí, ale často takový obrázek vložit nelze, protože to způsobí podivné chyby (poškodí se následující obrázek, ztratí se část textu, nebo i celý zbytek dokumentu). Expert někdy dokáže takový soubor opravit, ale může to vyžadovat několikahodinové studium. Naštěstí existují nástroje, které takovou nápravu zvládnou. Zmíníme se o nich v kapitole 7.5.

Další pověra se týká bitmapového formátu TIFF (Tagged Image File Format, soubory mají obvykle příponu tif). Obvykle se tvrdí, že tento formát používá bezztrátovou kompresi. Ve skutečnosti si však při ukládání můžeme kompresi zvolit. Máme na výběr několik bezztrátových algoritmů, dokonce lze obrázek uložit zcela bez komprese, ale můžeme též použít ztrátovou kompresi JPEG.

⁴Vyslovuje se *jay peg*.

7.2 Načtení obrázku do dokumentu

V kapitole 6 jsme se věnovali plovoucím prostředím a metodám, jak vložit plovoucí objekt do textu. Nyní se zaměříme na to, jak do plovoucího prostředí načteme vlastní obrázek.

TeX je program napsaný pro zpracování textu. Kromě primitivní grafiky tvořené vodorovnými a svislými úsečkami a znaky ze speciálně připravených fontů nelze žádné obrázky přímo vytvářet. Neexistuje ani příkaz pro vložení obrázku. TeX má ovšem příkaz `\special`, který není vůbec interpretován a jeho argument je pouze poslán výstupnímu zařízení. Záleží na příslušném ovladači, jak s takovým povelům naloží. Mluvíme-li o tom, jak se do TeXových či LaTeXových dokumentů vkládají obrázky, máme vždy na mysli nějaký konkrétní ovladač zařízení.

Working Papers České národní banky jsou zveřejňovány ve formátu PDF. Zaměříme se tedy na dvě nejběžnější metody, jak získat PDF z textu psaného v LaTeXu. První možností je konverze souboru DVI na PostScript programem `dvips`. Postscriptový soubor pak převedeme na PDF nějakým programem (existují komerční i volně šiřitelné programy). Druhou možností je přímá tvorba PDF programem `pdfTeX`.

Při použití programu `dvips` je optimálním grafickým formátem EPS, a to jak pro bitmapové, tak pro vektorové obrázky. Program `pdfTeX` umí načíst přímo obrázky ve formátech JFIF a PNG, ale ty jsou bitmapové a budeme je používat jen zřídka. Vektorové obrázky musíme připravit ve formátu PDF. Pro převod z EPS do PDF lze využít komerční Adobe Distiller. Stejnou funkci však nabízí i program `epstopdf`. Ten však vyžaduje GhostScript a perl. Zmíněné programy jsou volně šiřitelné a existují pro všechny operační systémy.

Obrázek lze načíst různými způsoby. Nejvhodnějším je zřejmě využití balíčku `graphics`. Makro `\includegraphics` má jeden povinný parametr, jímž je jméno souboru obsahujícího obrázek. V nepovinném parametru v hranatých závorkách lze uvést řadu rozličných pokynů ve tvaru *klíč=hodnota*. Syntaxe je stejná jako u makra `\typItem`, jež bylo popsáno v kapitole 5.2.2, pouze místo složených závorek používáme závorky hranaté, neboť první parametr je nepovinný. I zde jsou parametry zpracovány pomocí balíčku `keyval`.

Podrobný popis příkazu `\includegraphics` najdete v souboru `grfguide.ps`, jenž je součástí distribuce. Zde si popíšeme pouze několik základních parametrů:

scale je poměr zmenšení či zvětšení. Zmenšení na polovinu dosáhneme zadáním `scale=.5`.

width specifikujeme požadovanou šířku. Rozměr lze zadat v libovolných jednotkách, ale i jako podíl jiných rozměrů. Požadujeme-li šířku 75 % šířky textu, použijeme `width=.75\textwidth`.

height určuje výšku obrázku. Uvedeme-li současně `height` i `width`, dojde k distorzi obrázku. Použijeme-li pouze jeden z těchto parametrů, druhý z nich se dopočte tak, aby poměr stran zůstal zachován.

clip způsobí oříznutí na ohraničovací rámeček. Některé programy (např. Quatro) vytvářejí vadné EPS, jež vymažou své okolí. Pokud při vložení obrázku zmizí část textu nad ním nebo vedle něho, vyzkoušejte použití parametru `clip` (bez uvedení rovnítka a hodnoty).

page používáme pouze při vkládání obrázku z vícestránkového souboru PDF. Zadáváme jím pořadové číslo stránky, kterou chceme vložit.

7.3 Příprava vektorového obrázku

Připravujeme-li vektorový obrázek, musíme především použít grafický editor nebo obdobný program, který skutečně pracuje s vektorovou reprezentací. Obrázek pak musíme převést do formátu, který lze do dokumentu načíst, tj. EPS nebo PDF. Výhodou je, pokud program umí soubor v tomto formátu zapsat přímo. V opačném případě si musíme pomoci nějakou náhradní metodou.

V následujících podkapitolách si popíšeme několik běžných programů a ukážeme si úskalí, jež nás mohou potkat.

7.3.1 Gnuplot

Gnuplot je flexibilní volně šiřitelný program pro kreslení matematických grafů. Je dostupný pro všechny operační systémy. Byl jím vytvořen obrázek 1. Umožňuje výstup v mnoha formátech včetně EPS, nové verze i v PDF.

Program nabízí bohaté možnosti ovlivňování barvy křivek i bodů a tvaru bodů. Tyto možnosti závisí na zvoleném výstupním formátu, dokonce stejný typ křivky může být v různých výstupních formátech zobrazen jinou barvou. Informaci získáte nejlépe pomocí příkazu `test`, jíž získáte testovací obrazec představující všechny možnosti příslušného formátu. Chcete-li zjistit, jaké možnosti poskytuje výstup ve formátu EPS, použijte příkazy:

```
set term postscript eps color solid lw 2 18
set output "test.eps"
test
set output
quit
```

Testovací obrazec bude v souboru `test.eps` v aktuálním adresáři.

7.3.2 Corel Draw

Corel Draw je komerční editor vektorových obrázků. Nabízí funkci exportu do formátu EPS. Nesmíme však zapomenout, že Corel Draw obsahuje spoustu písem, jež nejsou dostupná na jiných počítačích. Při exportu do EPS je proto musíme převést do křivek. Počínaje verzí 8 již tuto akci program provede automaticky.

7.3.3 Adobe Illustrator

Adobe Illustrator je komerční vektorový editor, jehož nativní formát AI je založen na PostScriptu. Otevřeme-li soubor v běžném textovém editoru, na první pohled jej nerozeznáme od EPS. PostScript je však programovací jazyk, jenž má s \TeX em jeden společný rys: umožňuje definici uživatelských příkazů (operátorem `def`). Formát AI sice obsahuje postscriptové příkazy, ale pro jejich interpretaci je nutno mít nadefinována

makra, jež má Adobe Illustrator v sobě. Pokud vložíte do jiného dokumentu přímo obrázek ve formátu AI, vše bude zdánlivě fungovat, ale jen do chvíle, kdy se pokusíte dokument vytisknout. Na stránce, obsahující tento obrázek, interpret PostScriptu nahlásí chybu typu `undefined`. Nezapomeňte proto uložit obrázek ve formátu EPS.

7.4 Virtuální tiskárny

Virtuální tiskárny představují náhradní řešení pro případ, že program nemá výstup do formátu EPS. Metoda je ovšem použitelná pouze v případě, že používáme vektorový editor. Bitmapový grafický editor i při použití virtuální tiskárny vytvoří bitmapový EPS či PDF.

7.4.1 Tiskárny s výstupem do PostScriptu

Na trhu jsou dostupné tiskárny s podporou PostScriptu. Pro tyto tiskárny existují ovladače, jež jsou běžnou součástí instalačního CD operačního systému (Windows, Mac OS, OS/2, eComStation). Některé tiskárny podporují více komunikačních jazyků, jakými jsou PCL a HPGL. Použití takového ovladače ve funkci virtuální tiskárny není vhodné, neboť na začátku a na konci vygenerovaného souboru jsou obvykle příkazy jazyka PDL nebo obdobného, jimiž se zapíná a vypíná interpret jazyka PostScript. Takové příkazy nám pro další zpracování napáchají více škody než užitku. Nejlepší jsou většinou tiskárny Apple Writer, případně speciální virtuální ovladače získané z <http://www.adobe.com>. Chceme-li vytvářet barevné obrázky ve formátu EPS, musíme nainstalovat ovladač barevné tiskárny. Ovladač černobílé tiskárny totiž může převést barvy na stupně šedi.

Chceme-li nainstalovat virtuální tiskárnu ve Windows, je důležité z několika možností zvolit tu nejlepší. Tiskárnu instalujeme jako lokální a nepřipojíme ji k žádnému fyzickému zařízení, nýbrž k souboru. Pokud ovladač umožňuje nastavení funkcí, zapneme formát EPS, požadujeme konverzi fontů na Type 1 a jejich vkládání do dokumentu. Při tisku **nesmíme zaškrtnout tisk do souboru**, ale vyčkáme, až nás ovladač k zadání jména souboru sám vyzve.

7.4.2 Tiskárny s výstupem do PDF

Virtuální tiskárnu s výstupem do PDF nabízí komerční Adobe Acrobat. Existuje též řada sharewarových programů generujících PDF, jež se nainstalují jako virtuální tiskárny. Všechny mají svůj vlastní instalační program. Opět se snažíme, je-li to možné, zapnout konverzi fontů na Type 1 a vkládání fontů do dokumentu.

7.5 Oprava vadných souborů EPS a PDF

V kapitole 7.1, na straně 19, jsme se zmínili o tom, že obrázky v souborech EPS a PDF mohou být poškozeny. Nyní si popíšeme, jak je lze opravit.

První častou chybou je nesprávný či nevhodný ohraničovací rámeček (Bounding-Box). Ohraničovací rámeček označuje rozměry obrázku v souboru EPS. Podle specifikace musí být veškerý obrázek uvnitř rámečku, ale není řečeno, že musí být rámeček

těsný. Pokud je ohraničovací rámeček příliš velký, není to chyba, ale není to užitečné. Podle této informace totiž program vkládá obrázek do dokumentu. Je-li ohraničovací rámeček příliš velký, vynechá program okolo obrázku velké volné místo. Nápravy nejlépe dosáhneme volně šířitelným programem GhostScript a jeho pohodlnou nadstavbou GhostView. Potřebujeme oba programy, pro Linux existuje řada různých nadstavb (např. ggv pro Gnome). Přepneme orientaci obrázku na Portrait a zvolíme funkci s nepřiliš vhodným názvem PS to EPS. Funkce totiž nekonvertuje PostScript na EPS, pouze vygeneruje těsný ohraničovací rámeček se správnými rozměry. V naprosté většině případů funguje správně automatické nastavení. Ruční nastavení využijeme v případech, kdy automat selže, nebo v případech, kdy se chceme části obrázku zbavit a plánujeme použití parametru `clip` v makru `\includegraphics`.

! GhostView neumí zjistit potřebnou velikost papíru pro zobrazení obrázku ve formátu EPS. Navíc některé programy, zejména virtuální tiskárny, nevloží obrázek do levého dolního rohu. Pokud vidíte pouze prázdnou stránku, zkuste v menu Media zvolit větší velikost papíru.

Rozměry stránky v souboru PDF změníme pomocí plného Adobe Acrobatu, jenž tuto funkci nabízí. Neznám volně šířitelnou alternativu.

Horší situace nastane, jestliže se soubor pouze tváří jako EPS. Nejjednodušší je případ, kdy na začátku a na konci souboru máme příkazy PDL nebo jiného jazyka. GhostView si s nimi obvykle poradí, ale problémy vzniknou při vkládání obrázku. Náprava je snadná. Soubor otevřeme v obyčejném textovém editoru a vymažeme vše před prvním výskytem znaků `%!PS` (v EPS musí být na začátku prvního řádku) a vše za textem `%%EOF`. Při základní znalosti PostScriptu lze napravit i další chyby, obvykle stačí vymazat zakázané příkazy, ale tato úloha může být i pro experta značně obtížná. Nejpohodlnější je převedení takového obrázku na PDF, buď Adobe Distillerem, nebo programem `ps2pdf` obsaženým v distribuci GhostScriptu (případně programem `eps2pdf`). Pokud nechceme nebo nemůžeme zpracovat dokument `pdfTeXem`, převedeme PDF zpět na EPS. Použijeme buď plný Adobe Acrobat (funkce Save As EPS), nebo program `pdftops` s parametrem `-eps`. Program `pdftops` je součástí volně šířitelného `xpdf`, viz <http://www.foolabs.com/xpdf/>. GhostScript též nabízí konverzi PDF na EPS, ale bohužel přitom vyrastruje fonty. Takový obrázek pak ovšem není použitelný.

8 Příprava tabulek

Tabulky jsou důležitou součástí odborných článků. Užitečné však mohou být jen v případě, že jsou uspořádány přehledně. Nad tvorbou tabulek obvykle strávíme nejdelší část přípravy dokumentu. V této kapitole si předvedeme několik technik, jež práci při sazbě tabulek usnadní a umožní dosažení požadovaného vzhledu.

8.1 Zarovnání sloupce na desetinnou tečku

V tabulkách často používáme desetinná čísla, přičemž počet číslic v jednotlivých číslech téhož sloupce se může lišit. Přitom je nutné, aby čísla byla zarovnána na desetinnou tečku. Prostředí `tabular` takovou možnost nenabízí, ale řešení lze nalézt v použití balíčku `dcolumn`. Tento balíček přidává do prostředí `tabular` sloupec typu D. Specifikace sloup-

ce typu D vyžaduje tři parametry: separátor použitý ve zdrojovém textu, separátor, jenž má být vytištěn a maximální počet desetinných míst. Bude-li poslední parametr záporný, bude desetinná tečka uprostřed sloupce, což může vést k tomu, že sloupec bude příliš široký. První dva parametry budou většinou shodné, ale nemusí to tak být nutně. Předpokládejme, že jsme tabulku exportovali z nějakého tabulkového programu, kde jsou používány desetinné čárky. My však chceme mít desetinné tečky. Abychom nemuseli zasahovat do souboru, využijeme možnosti nabízené prvními dvěma parametry. Příklad najdete v tabulce 1 a zdrojový kód příkladu na obrázku 2.

Tabulka 1: Demonstrace tabulky se zarovnáním na desetinnou tečku

Jméno	Výška [m]	Váha [kg]
Zlatovláska	1.68	62.3
Dlouhý	12.6	98.1
Široký	1.83	386
Bystrozraký	1.74	74.2

Obrázek 2: Zdrojový kód tabulky 1

```

\setlength\extrarowheight{2pt}
\begin{tabular}{|l|D{,}{.}{2}D{,}{.}{1}|}\hline
\bfseries Jm\`eno & \multicolumn{1}{c}{\bfseries V\`yška [m]} &
\multicolumn{1}{c}{\bfseries V\`aha [kg]}\hline
Zlatovl\`aska & 1,68 & 62,3 \\\
Dlouh\`y & 12,6 & 98,1 \\\
\v{S}irok\`y & 1,83 & 386 \\\
Bystrozrak\`y & 1,74 & 74,2 \\\hline
\end{tabular}

```

Nadpis ve sloupci typu D by nebyl umístěn správně. Musíme proto použít makro `\multicolumn`, v němž dané buňce změníme zarovnání. Všimněte si, že součástí specifikace zarovnání je **následující** svislá čára, nikoliv předcházející. Tu musíme uvádět pouze u prvního sloupce. Pokud by nadpis *Jméno* měl být z nějakého (v tomto případě nerozumného) důvodu zarovnán vpravo, museli bychom použít:

```
\multicolumn{1}{|r|}{\bfseries Jm\`eno}
```

Definice nového typu sloupce je umožněna tím, že je implicitně zaveden balíček `array`. Ten uživatelům nabízí makro `\newcolumntype`, jímž se nové typy definují. Syntaxe se podobá makru `\newcommand` s tím rozdílem, že definujeme typ sloupce, nikoliv makro, a nelze deklarovat nepovinný parametr s defaultní hodnotou. Kdybychom si v příkladu z obrázku 2 definovali nový typ sloupce:

```
\newcolumntype{d}[1]{D{,}{.}{#1}}
```


mohli jsme preambuli tabulky zapsat přehledněji:

```
\begin{tabular}{|l|d{2}d{1}|}
```

V tabulce jsme nastavili ještě rozměr `\extrarowheight`, jenž je definován též v balíčku `array`. Písmena s českými diakritickými znaménky jsou moc vysoká a mezi textem a linkou by nebyla téměř žádná mezera. Rozměrový registr `\extrarowheight` zvětšuje mezeru mezi textem a linkou nad ním. Parametr v hranatých závorkách `makra` `\` zvětšuje pouze mezeru pod textem.

8.2 Široké tabulky

Některé tabulky mohou být velmi široké. Jedinou možností, jak je vytisknout, je otočení o 90° . Docílíme toho použitím balíčku `rotating` a prostředím `sidewaystable`. (Analogicky lze otočit i obrázky v prostředí `sidewaysfigure`.) Balíček `rotating` bohužel otáčí tabulky jinak na lichých a jinak na sudých stránkách, což je z typografického hlediska nepřijatelné. Abychom tento problém odstranili, musíme balíček zavést v preambuli následujícím způsobem:

```
\usepackage[figuresright]{rotating}
```

Ukázku široké tabulky, jež obsahuje matici náhodných čísel vypočtenou v programu Octave příkazem `5 × randn(20, 10)`, vidíte v tab. 2, zdrojový kód najdete na obrázku 3. Tabulka bude otočena tak, že levý okraj bude umístěn na spodní okraj sazebního obrazce (tj. zarovnan se spodním okrajem textu na jiných stránkách). To jen zřídka působí esteticky. Lépe vypadá vycentrovaná tabulka, čehož dosáhneme příkazem `\centering`. Je nutno též vycentrovat popisek explicitním uvedením `\centeredcaption` místo `\caption`.

Centrování tabulky nemusí být vždy žádoucí. Vzhled stránky lze upravit posunem. Stránku připravíme bez centrování a vytiskneme, nebo zobrazíme v GhostView (tento program umožňuje odměřování). Předpokládejme, že chceme tabulku posunout o 27 mm směrem k hornímu okraji stránky. Musíme tedy o tuto velikost zvětšit levý okraj plovoucího prostředí. Zařídíme to tím, že hned na jeho počátku nastavíme:

```
\begin{sidewaystable}[p]% začátek otočené tabulky  
\setlength{\leftskip}{27mm}
```

Tento příkaz posune tabulku i standardní popisek, ale nebude správně fungovat při použití `\centeredcaption`.

! Všimněte si, že příkaz `\label` musí být uveden uvnitř plovoucího prostředí, k němuž se vztahuje, a to až za makrem `\caption` nebo jeho alternativou definovanou v třídě `cnbwp`. Pokud jej uvedete mimo toto prostředí nebo před makrem `\caption`, bude se vztahovat k `\section` či `\subsection`, v níž se vyskytuje.

9 Předání dokumentu ke korektuře a ke zveřejnění

Pravidla pro předávání dokumentů a k provádění korektur mohou být upravena Českou národní bankou. V této kapitole budou uvedeny obecné instrukce vycházející ze

Tabulka 2: Široká tabulka

#	Col. 1	Col. 2	Col. 3	Col. 4	Col. 5	Col. 6	Col. 7	Col. 8	Col. 9	Col. 10
Row 1	-6.412	-2.654	-5.300	4.358	-6.473	-4.573	2.391	-0.497	-4.262	-0.341
Row 2	2.799	-8.109	5.647	-0.214	4.665	2.971	13.699	-5.059	-0.088	4.090
Row 3	8.427	5.467	-7.061	-0.347	-6.955	6.352	-3.955	7.768	-9.852	4.618
Row 4	-1.978	-1.226	1.136	1.733	3.874	15.072	4.112	-1.931	4.127	1.177
Row 5	10.896	6.859	-0.623	6.685	-6.378	2.714	-2.670	7.862	-2.314	-4.094
Row 6	-5.013	-2.391	-1.763	-1.499	-4.053	2.453	1.550	-3.939	3.366	-0.780
Row 7	8.644	-1.787	-5.782	1.244	-0.806	3.506	1.810	-3.908	-0.626	-1.933
Row 8	-4.965	-2.494	1.539	6.265	0.892	-2.730	3.311	-0.006	3.735	0.408
Row 9	1.237	-3.029	-0.773	9.400	-6.009	-0.487	4.281	4.520	-5.744	-3.628
Row 10	0.051	-8.717	-1.366	1.811	-1.599	-10.179	-1.355	6.024	4.912	0.728
Row 11	1.894	-8.089	-6.445	-9.112	-4.753	2.555	2.751	0.952	-0.291	-1.523
Row 12	-3.103	-0.002	2.733	-9.805	-3.154	-1.985	4.259	-2.340	-2.236	-2.372
Row 13	3.729	1.978	6.627	-9.898	3.746	-3.595	-6.425	10.043	4.578	7.770
Row 14	9.511	-8.231	1.815	-5.189	-1.213	0.767	-2.620	6.613	-1.119	-3.838
Row 15	-0.699	-10.599	5.787	-11.333	-4.810	2.769	0.255	-6.831	-1.643	-2.870
Row 16	7.190	-2.291	7.532	2.650	-5.878	-4.859	7.792	-1.337	-5.075	-7.241
Row 17	-5.918	0.987	5.037	-0.556	-2.653	-7.008	3.491	-1.028	0.573	4.620
Row 18	-3.957	-4.265	1.325	3.102	-5.731	-3.944	-6.565	5.178	2.477	-1.948
Row 19	-1.228	-0.170	-3.048	-2.966	9.791	9.006	9.186	-2.971	8.657	-2.838
Row 20	-2.340	-4.932	-3.904	4.164	-5.838	-7.320	1.451	4.955	7.439	-4.407

Obrázek 3: Zdrojový kód tabulky 2

```

\begin{sidewaystable}[p]
\setlength{\extrarowheight}{2pt}
\newcommand\mcol[1]{\multicolumn{1}{r}{Col. #1}}
\centeredcaption{\v{S}irok'a tabulka\label{widetable}\bigskip\centering}
\begin{tabular}{l*{10}{D{.}{.}{3}}}\hline
\bfseries \# & \mcol{1}& \mcol{2}& \mcol{3}& \mcol{4}&
& \mcol{5}& \mcol{6}& \mcol{7}& \mcol{8}& \mcol{9}& \mcol{10}\\\hline
Row 1 & -6.412 & -2.654 & -5.300 & 4.358 & -6.473 & -4.573 & 2.391 & -0.497 & -4.262 & -0.341\\
Row 2 & 2.799 & -8.109 & 5.647 & -0.214 & 4.665 & 2.971 & 13.699 & -5.059 & -0.088 & 4.090\\
Row 3 & 8.427 & 5.467 & -7.061 & -0.347 & -6.955 & 6.352 & -3.955 & 7.768 & -9.852 & 4.618\\
Row 4 & -1.978 & -1.226 & 1.136 & 1.733 & 3.874 & 15.072 & 4.112 & -1.931 & 4.127 & 1.177\\
Row 5 & 10.896 & 6.859 & -0.623 & 6.685 & -6.378 & 2.714 & -2.670 & 7.862 & -2.314 & -4.094\\
Row 6 & -5.013 & -2.391 & -1.763 & -1.499 & -4.053 & 2.453 & 1.550 & -3.939 & 3.366 & -0.780\\
Row 7 & 8.644 & -1.787 & -5.782 & 1.244 & -0.806 & 3.506 & 1.810 & -3.908 & -0.626 & -1.933\\
Row 8 & -4.965 & -2.494 & 1.539 & 6.265 & 0.892 & -2.730 & 3.311 & -0.006 & 3.735 & 0.408\\
Row 9 & 1.237 & -3.029 & -0.773 & 9.400 & -6.009 & -0.487 & 4.281 & 4.520 & -5.744 & -3.628\\
Row 10 & 0.051 & -8.717 & -1.366 & 1.811 & -1.599 & -10.179 & -1.355 & 6.024 & 4.912 & 0.728\\
Row 11 & 1.894 & -8.089 & -6.445 & -9.112 & -4.753 & 2.555 & 2.751 & 0.952 & -0.291 & -1.523\\
Row 12 & -3.103 & -0.002 & 2.733 & -9.805 & -3.154 & -1.985 & 4.259 & -2.340 & -2.236 & -2.372\\
Row 13 & 3.729 & 1.978 & 6.627 & -9.898 & 3.746 & -3.595 & -6.425 & 10.043 & 4.578 & 7.770\\
Row 14 & 9.511 & -8.231 & 1.815 & -5.189 & -1.213 & 0.767 & -2.620 & 6.613 & -1.119 & -3.838\\
Row 15 & -0.699 & -10.599 & 5.787 & -11.333 & -4.810 & 2.769 & 0.255 & -6.831 & -1.643 & -2.870\\
Row 16 & 7.190 & -2.291 & 7.532 & 2.650 & -5.878 & -4.859 & 7.792 & -1.337 & -5.075 & -7.241\\
Row 17 & -5.918 & 0.987 & 5.037 & -0.556 & -2.653 & -7.008 & 3.491 & -1.028 & 0.573 & 4.620\\
Row 18 & -3.957 & -4.265 & 1.325 & 3.102 & -5.731 & -3.944 & -6.565 & 5.178 & 2.477 & -1.948\\
Row 19 & -1.228 & -0.170 & -3.048 & -2.966 & 9.791 & 9.006 & 9.186 & -2.971 & 8.657 & -2.838\\
Row 20 & -2.340 & -4.932 & -3.904 & 4.164 & -5.838 & -7.320 & 1.451 & 4.955 & 7.439 & -4.407\\
\hline
\end{tabular}
\end{sidewaystable}

```

zdravého rozumu a ze zkušeností se sdílením dokumentů mezi počítači s různými operačními systémy v multilinguálním prostředí.

9.1 Příprava dokumentu k odevzdání

Při přípravě dokumentu k odevzdání je nutno mít na zřeteli, že jazykový korektor nemusí být expertem na \LaTeX , a dokonce vůbec nemusí mít žádnou distribuci \TeX u nainstalovanou. Je tedy nutno odevzdat soubor, který lze vytisknout běžnými snadno instalovatelnými programy, nejlépe tedy ve formátu PDF, v nouzi jako PostScript.

Při psaní dokumentu omezte počet příkazů `\input` a `\include`. Jejich nadměrné užívání vede spíše k nižší přehlednosti.

V dokumentech nepoužívejte písmena s diakritickými znaménky. Je příliš mnoho kódování češtiny a slovenštiny a různé distribuce \TeX u nakládají s kódováním různě. Počítejte spíše s tím, že s dokumentem bude pracovat mírně poučený laik, jenž si s překódováním nemusí vědět rady. Diakritická znaménka tedy zadávejte vždy pomocí \TeX ových sekvencí.

Všechny soubory, nutné pro sazbu dokumentu, musí být v jednom adresáři, nebo (raději výjimečně) v podadresářích adresáře s hlavním dokumentem. Příkaz

```
\includegraphics{d:/projects/figures/image1.eps}
```


je špatný hned ze dvou důvodů:

1. Autor dokumentu může zapomenout, že nezbytný soubor je v jiném adresáři, a zapomene jej přibalit.
2. Osoba, zpracovávající dokument, může mít jinak rozdělený diskový prostor a příkaz pak bude hlásit, že požadovaný soubor nebyl nalezen.

Uživatelé unixových systémů by měli odolat poušení použít v takových případech symbolické linky. Jednak se snadno stane, že k dokumentu nepřibalí soubory, ale jen symbolické linky, a kromě toho musí být dokument zpracovatelný i na souborovém systému, který symbolické linky nepodporuje. Použijete-li podadresáře, zadávejte jména relativně, např.:

```
\includegraphics{figures/image1.eps}
```

Vždy však uvažte, zda je vytváření podadresářů skutečně nezbytné. Uložení všech příbuzných souborů v jednom adresáři je obvykle pro nezavěšeného člověka přehlednější.

 The \TeX book uvádí, že jméno souboru je ukončeno znakem mezera. Vzhledem k oblíbenosti mezer firmou Microsoft existují distribuce, jež si s tím dovedou poradit. Je to však **velmi nestandardní rozšíření**, které nemusí fungovat ani na témže operačním systému s jinou distribucí \TeX u. Ještě horší problém představují názvy adresářů a souborů obsahující písmena s diakritickými znaménky. Jejich funkčnost závisí na příliš mnoha faktorech, nejen na distribuci \TeX u, ale i na nastavení locales v operačním systému. Použití takových jmen je nutno se vyhnout, neboť s vysokou pravděpodobností způsobí problémy.

Všechny potřebné soubory je nutno zabalit do jediného archivu, a to i s adresářovou strukturou. Optimální je formát ZIP. Zejména v případě, kdy se dokument skládá z mnoha souborů, přidáme krátký informativní text v souboru pojmenovaném `readme.txt`.

9.2 Provádění korektur


Základní typografické pravidlo říká, že korektury se nikdy neprovádějí na obrazovce počítače, vždy se provádějí na vytištěném textu na papíře. Prvním krokem korektury je tedy vytištění dokumentu dodaného ve formátu PDF či PostScript.

Dokumenty určené pro zpracování v \LaTeX u jsou obyčejné textové soubory. Na jejich editaci nepotřebujeme žádný specializovaný program, ty jen ulehčují práci. Ve Windows je můžeme otevřít např. v programu `notepad`.

\TeX byl vytvořen v době, kdy se texty do počítače zadávaly na děrných štítcích. Proto nezáleží na tom, jak je text rozdělen na řádky a kolik mezer je mezi slovy. Důležité je, aby mezi odstavci byl vynechán prázdný řádek.

Hlavní dokument může načítat jiné soubory pomocí příkazů `\input` nebo `\include`, přičemž jméno načítaného souboru je zadáno v parametru tohoto makra. Příponu `.tex` není nutno uvádět.

Formátování textu se ovlivňuje příkazy, jež začínají zpětným lomítkem. Některé z nich mají argumenty zapsané ve složených, někdy v hranatých závorkách. Tyto příkazy nesmí být při korektuře změněny.

 Korektor smí opravovat pouze slova, která vidí ve vytištěné verzi. Nesmí zasahovat do formátovacích značek, neboť by to mohlo vážně poškodit dokument při následném zpracování. Nemá-li korektor jistotu, zda smí něco v dokumentu opravit, je lepší, když opravu označí na papíře nebo v samostatném souboru a o její zanesení požádá autora.

Jako zajímavá alternativa se nabízí provádění korektur v souboru PDF. Korektury je nutno povolit pomocí plné verze programu Adobe Acrobat, pro vlastní zápis korektur postačí Acrobat Reader.

10 Vzorové soubory

Součástí tohoto manuálu je několik vzorových souborů. Hlavní šablona dokumentu je v souboru `cnbpaper.tex` a celý dokument je převeden do formátu PDF (soubor `cnbpaper.pdf`) pomocí `pdf \LaTeX` u. Soubor ovšem lze zpracovat i běžným \LaTeX em a programem `dvips`.

Některé ukázky jsou uloženy v samostatných souborech, jednak proto, aby byla zaručena jejich konzistence s tímto návodem, a jednak proto, aby uživatel při psaní svého textu nemusel pracovat s dlouhým vzorovým dokumentem.

Tabulka 1, předvádějící zarovnání na desetinnou tečku, se nachází v souboru `numtable.tex`. V souboru není uveden příkaz `\caption` ani plovoucí prostředí. Tyto příkazy najdete v hlavní šabloně dokumentu.

Širokou tabulku 2 najdete v souboru `widematrix.tex`. V tomto případě soubor obsahuje též příkazy pro definici prostředí.

Soubory graf18.eps a graf18.pdf jsou vektorové grafy z obrázku 1. Šablona používá balíček ifpdf pro zjištění, zda je používán pdf \LaTeX , a podle toho bude načten soubor ve správném formátu. Pokud není v distribuci balíček ifpdf obsažen, předpokládá se, že pdf \LaTeX není k dispozici.

Soubor biblio.tex obsahuje seznam literatury zapsaný pomocí maker z kapitoly 5.2. Celý výpis je uveden v Příloze.

Soubor cnbsample.bib obsahuje bibliografickou databázi pro Bib \TeX . Z tohoto souboru byl vygenerován výše zmíněný soubor biblio.tex.

Pokud budete používat soubor cnbpaper.tex jako šablonu svého dokumentu, zamyslete se nad tím, které balíčky skutečně potřebujete. Nepotřebné příkazy `\usepackage` vymažte. Je možné, že budete potřebovat nějaké další balíčky. Jedním z nich může být `amsmath`.

11 Změny, verze 2013.12

V prosinci 2013 byly provedeny tyto změny:

1. Zrušeny přepínače 11pt a 12pt pro nastavení velikosti písma v dokumentu.
2. Změna způsobu zadávání autorů, změna syntaxe makra `\author`.
3. Zrušena makra `\shortauthor` a `\shorttitle`
4. Změna použití makra `\acknowledge`.
5. Přidáno prostředí abstrakt pro český abstrakt.
6. Automaticky zaveden balíček babel pro aktivaci českých vzorů dělení v prostředí abstrakt.
7. Úpravy formátování nadpisů podle nových požadavků.
8. Implementována nová makra `\Note` a `\Source`.
9. Upraveno formátování jmen autorů v bibliografickém stylu `abbrvcnb`.
10. Balíček times nahrazen balíčkem mathptmx.
11. Aktualizován manuál.

A Příloha

V příloze je uveden příklad všech typů prací zapsaných pomocí maker z kapitoly 5.2 a jejích podkapitol. Položku `note` obvykle používat nebudete.

Celý příklad je dostupný též v souboru biblio.tex.

```
% Requires cnbwp version 2005/12/23 or newer
```

```
\begin{thebibliography}{29}
\manualItem[man()]{manual-minimal}{
  title = {The Definitive Computer Manual},
  note = {This is a minimal MANUAL entry},
```

```

}

\proceedingsItem[pro(1983)]{proceedings-minimal}{
  title = {Proc. Fifteenth Annual Symposium on the Theory of Computing},
  year = {1983},
}

\articleItem[Aamport(1986b)]{article-full}{
  author = {Aamport, L.~A.},
  title = {The Gnats and Gnus Document Preparation System},
  journal = {\mbox{G-Animal's} Journal},
  year = {1986b},
  volume = {41},
  number = {7},
  pages = {73+},
  month = {Jul},
  note = {This is a full ARTICLE entry},
}

\articleItem[Aamport(1986a)]{article-minimal}{
  author = {Aamport, L.~A.},
  title = {The Gnats and Gnus Document Preparation System},
  journal = {\mbox{G-Animal's} Journal},
  year = {1986a},
}

\bookItem[Knuth(1981c)]{book-full}{
  author = {Knuth, D.~E.},
  title = {Seminumerical Algorithms},
  publisher = {Addison-Wesley},
  year = {1981c},
  volume = {2},
  series = {The Art of Computer Programming},
  address = {{Reading, Massachusetts}},
  edition = {second},
  month = {10~Jan},
  note = {This is a full BOOK entry},
}

\bookItem[Knuth(1981)]{book-minimal}{
  author = {Knuth, D.~E.},
  title = {Seminumerical Algorithms},
  publisher = {Addison-Wesley},
  year = {1981},
}

\inbookItem[Knuth(1973a)]{inbook-full}{
  author = {Knuth, D.~E.},

```

```

title = {Fundamental Algorithms},
chapter = {1.2},
pages = {10--119},
publisher = {Addison-Wesley},
year = {1973a},
volume = {1},
series = {The Art of Computer Programming},
type = {section},
address = {{Reading, Massachusetts}},
edition = {second},
month = {10~Jan},
note = {This is a full INBOOK entry},
}

```

```

\inbookItem[Knuth(1973)]{inbook-minimal}{
author = {Knuth, D.~E.},
title = {Fundamental Algorithms},
chapter = {1.2},
publisher = {Addison-Wesley},
year = {1973},
}

```

```

\bookletItem[Knvth(1988)]{booklet-full}{
title = {The Programming of Computer Art},
author = {Knvth, J.~C.},
howpublished = {Vernier Art Center},
address = {{Stanford, California}},
month = {Feb},
year = {1988},
note = {This is a full BOOKLET entry},
}

```

```

\incollectionItem[Lincoll(1977b)]{incollection-full}{
author = {Lincoll, D.~D.},
title = {Semigroups of Recurrences},
booktitle = {High Speed Computer and Algorithm Organization},
publisher = {Academic Press},
year = {1977b},
editor = {Lipcoll, D.~J., D.~H. Lawrie, and A.~H. Sameh},
number = {23},
series = {Fast Computers},
type = {part},
chapter = {3},
pages = {179--183},
address = {New York},
edition = {third},
month = {Sep},
note = {This is a full INCOLLECTION entry},
}

```


}

```
\incollectionItem[Lincolll(1977a)]{incollection-minimal}{  
  author = {Lincolll, D.~D.},  
  title = {Semigroups of Recurrences},  
  booktitle = {High Speed Computer and Algorithm Organization},  
  publisher = {Academic Press},  
  year = {1977a},  
}
```

```
\bookItem[Lipcoll et~al.(1977c)Lipcoll, Lawrie, and Sameh]{whole-collection}{  
  editor = {Lipcoll, D.~J., D.~H. Lawrie, and A.~H. Sameh},  
  title = {High Speed Computer and Algorithm Organization},  
  publisher = {Academic Press},  
  year = {1977c},  
  number = {23},  
  series = {Fast Computers},  
  address = {New York},  
  edition = {third},  
  month = {Sep},  
  note = {This is a cross-referenced BOOK (collection) entry},  
}
```

```
\manualItem[Manmaker(1986)]{manual-full}{  
  title = {The Definitive Computer Manual},  
  author = {Manmaker, L.},  
  organization = {Chips-R-Us},  
  address = {Silicon Valley},  
  edition = {silver},  
  month = {Apr-May},  
  year = {1986},  
  note = {This is a full MANUAL entry},  
}
```

```
\manualItem[Manmaker(1984)]{manual-question}{  
  title = {Are Computer Manuals Useful?},  
  author = {Manmaker, L.},  
  organization = {Chips-R-Us},  
  year = {1984},  
}
```

```
\mastersthesisItem[Masterly(1988b)]{mastersthesis-full}{  
  author = {Masterly, {'E}.},  
  title = {Mastering Thesis Writing},  
  school = {Stanford University},  
  year = {1988b},  
  type = {Master's project},  
  address = {English Department},
```

```

month = {Jun-Aug},
note = {This is a full MASTERSTHESIS entry},
}

```

```

\mastersthesisItem[Masterly(1988a)]{mastersthesis-minimal}{
author = {Masterly, {\'E}.},
title = {Mastering Thesis Writing},
school = {Stanford University},
year = {1988a},
}

```

```

\miscItem[Missilany(1984)]{misc-full}{
author = {Missilany, J.-B.},
title = {Handing out random pamphlets in airports},
howpublished = {Handed out at O'Hare},
month = {Oct},
year = {1984},
note = {This is a full MISC entry},
}

```

```

\inproceedingsItem[Oaho et~al.(1983a)Oaho, Ullman, and
Yannakakis]{inproceedings-minimal}{
author = {Oaho, A.~V., J.~D. Ullman, and M.~Yannakakis},
title = {On Notions of Information Transfer in VLSI Circuits},
booktitle = {Proc. Fifteenth Annual Symposium on the Theory of Computing},
year = {1983a},
}

```

```

\inproceedingsItem[Oaho et~al.(1983b)Oaho, Ullman, and
Yannakakis]{inproceedings-full}{
author = {Oaho, A.~V., J.~D. Ullman, and M.~Yannakakis},
title = {On Notions of Information Transfer in VLSI Circuits},
booktitle = {Proc. Fifteenth Annual Symposium on the Theory of Computing},
year = {1983b},
editor = {Oz, W.~V. and M.~Yannakakis},
number = {17},
series = {All ACM Conferences},
pages = {133--139},
address = {Boston},
month = {Mar},
organization = {The OX Association for Computing Machinery},
publisher = {Academic Press},
note = {This is a full INPROCEEDINGS entry},
}

```

```

\unpublishedItem[Overwood et~al.(.)Overwood, {\~N}et, and
\={P}ot]{unpublished-minimal2}{
author = {Overwood, U., N.~{\~N}et, and P.~\={P}ot},
}

```

```
title = {Lower Bounds for Wishful Research Results},
}
```

```
\unpublishedItem[Overwood et~al.(1988)Overwood, {\~N}et, and
\={P}ot]{unpublished-full2}{
author = {Overwood, U., N.~{\~N}et, and P.~\={P}ot},
title = {Lower Bounds for Wishful Research Results},
month = {{Nov, Dec}},
year = {1988},
}
```

```
\proceedingsItem[Oz and Yannakakis(1983)]{proceedings-full}{
title = {Proc. Fifteenth Annual Symposium on the Theory of Computing},
year = {1983},
editor = {Oz, W.~V. and M.~Yannakakis},
number = {17},
series = {All ACM Conferences},
address = {Boston},
month = {Mar},
organization = {The OX Association for Computing Machinery},
publisher = {Academic Press},
note = {This is a full PROCEEDINGS entry},
}
```

```
\phdthesisItem[Phony-Baloney(1988b)]{phdthesis-full}{
author = {Phony-Baloney, F.~P.},
title = {Fighting Fire with Fire: Festooning French Phrases},
school = {Fanstord University},
year = {1988b},
type = {PhD Dissertation},
address = {Department of French},
month = {Jun-Aug},
note = {This is a full PHDTHESIS entry},
}
```

```
\phdthesisItem[Phony-Baloney(1988a)]{phdthesis-minimal}{
author = {Phony-Baloney, F.~P.},
title = {Fighting Fire with Fire: Festooning French Phrases},
school = {Fanstord University},
year = {1988a},
}
```

```
\techreportItem[T'\{e}rrific(1988)]{techreport-full}{
author = {T'\{e}rrific, T.},
title = {An  $\{O(n \log n / \! \log \log n)\}$  Sorting Algorithm},
institution = {Fanstord University},
year = {1988},
type = {Wishful Research Result},
}
```

```

    number = {7},
    address = {{Computer Science Department, Fanstord, California}},
    month = {Oct},
    note = {This is a full TECHREPORT entry},
}

\techreportItem[Terrific(1988)]{techreport-minimal}{
    author = {Terrific, T.},
    title = {An  $\Theta(n \log n / \log \log n)$  Sorting Algorithm},
    institution = {Fanstord University},
    year = {1988},
}

\unpublishedItem["{U}nderwood et~al.()\ "{U}nderwood, {\~N}et, and
\={P}ot]{unpublished-minimal}{
    author = {"{U}nderwood, U., N.~{\~N}et, and P.~\={P}ot},
    title = {Lower Bounds for Wishful Research Results},
    note = {Talk at Fanstord University (this is a minimal UNPUBLISHED entry)},
}

\unpublishedItem["{U}nderwood et~al.(1988)\ "{U}nderwood, {\~N}et, and
\={P}ot]{unpublished-full}{
    author = {"{U}nderwood, U., N.~{\~N}et, and P.~\={P}ot},
    title = {Lower Bounds for Wishful Research Results},
    note = {Talk at Fanstord University (this is a full UNPUBLISHED entry)},
    month = {{Nov, Dec}},
    year = {1988},
}

\miscItem[Wagner()]{url-demo}{
    author = {Wagner, Z.},
    title = {Internet Home Page},
    note = {\url{http://icebearsoft.euweb.cz}},
}

\end{thebibliography}

```

B Index

Symbols	<code>\acknowledge</code> 8	<code>amsmath</code> 30
<code>\-</code> 13	<code>Acrobat</code> 22, 23, 29	<code>array</code> 24, 25
A	<code>Acrobat Reader</code> 29	<code>\articleItem</code> 11
<code>abbrvcnb</code> 10	<code>address</code> 13	<code>author</code> 13
<code>abbrvnat</code> 10	<code>Adobe Acrobat</code> . . 22, 23, 29	<code>\author</code> 8
<code>abstract</code> 8	<code>Adobe Distiller</code> 20, 23	B
<code>abstrakt</code> 8	<code>Adobe Illustrator</code> 21	<code>babel</code> 6
	<code>AI</code> 21	

