
Úprava dokumentů

texty k výuce základů typografie

Mgr. Šárka Vavrečková
sarka.vavreckova@fpf.slu.cz
<http://fpf.slu.cz/~vav10ui>

*Poslední změny:
25. dubna 2007*

Předmluva

Tento text je praktickým úvodem do problematiky správné úpravy dokumentů. Byl vytvářen jako jakási „kuchařka“ – postupy jsou spíše konkrétní a často doprovázené obrázky, vzadu v seznamu literatury nalezneme odkazy na zdroje dalších informací. Samotný dokument je ukázkou toho, jak by obdobná publikace mohla vypadat.

Dokument je určen především současným a budoucím studentům, které čeká psaní některého typu rozsáhlejší práce na vysoké škole. Předpokládá se určitá základní znalost práce alespoň s programem MS Word, ale přesto zde čtenář nalezne i mnohé základní postupy doprovázené obrázky a podrobněji popsané.

Text není třeba číst od začátku do konce, už proto, že některé pasáže jsou zřejmě zkušenějším uživatelům probíraných systémů známy. Hlavní struktura dokumentu se odvíjí od systému, ve kterém chce čtenář pracovat, tedy lze zvolit kapitolu pro daný systém. Typografické zásady jsou však vysvětlovány především v první kapitole věnované programu MS Word.

V textu jsou použity následující styly vyznačení:

- *kurzíva* vyznačuje důležité pojmy,
- bezpatkové písmo je využito pro vyznačení nápisů (položky menu, nadpisy dialogových oken apod.),
-  představuje klávesu na klávesnici,
- [modré neproporcionální písmo](#) určuje hypertextový odkaz na WWW stránku,
- neproporcionální písmo značí příkazy $\text{\TeX}$ a $\text{\LaTeX}$, textovou komunikaci s programy a přípony či názvy souborů,
- příklady jsou uvozeny slovem **Příklad** a číslem příkladu, úlohy slovem **Úlohy** a jsou ohraničeny shora i zdola čarou,
- **Výpis č. 0.1**

Takto je zformátován výpis kódu v $\text{\TeX}$ a $\text{\LaTeX}$.

Při sazbě dokumentu byl použit systém Mik $\text{\TeX}$ verze 2.5 a editor $\text{\TeX}$ nicCenter verze 7.01 a na této konfiguraci jsou také odzkoušeny postupy v kapitole 2 o ty-

pografickém systému $\text{\TeX}$. Postupy jsou kromě $\text{\TeX}$ u ukazovány na kancelářských balících MS Office XP a OpenOffice.org 2.0. Zde jsou jisté rozdíly mezi vlastnostmi různých verzí, na nejdůležitější z nich bude v textu upozorněno.

Nikdo není neomylný, ani já ne, je možné, že se v textu budou vyskytovat chyby. Proto přivítám upozornění na objevené chyby a také případné připomínky k samotnému textu.

Šárka Vavrečková

V Opavě dne 7. 11. 2006

OBSAH

Úvod	1
1 Typografie a MS Word	2
1.1 Uživatelská nastavení programu MS Word	2
1.2 Písmo	4
1.2.1 Nastavení písma	4
1.2.2 Zvláštní znaky	5
1.2.3 Vyznačování	7
1.3 Odstavcové styly	9
1.3.1 Formát odstavce	9
1.3.2 Seznamy	10
1.4 Styly a šablony	12
1.4.1 Vhodné nastavení stylů	12
1.4.2 Postup vytvoření hierarchie stylů	13
1.5 Struktura dokumentu	17
1.5.1 Co vše má dokument obsahovat	17
1.5.2 Textová část dokumentu	20
1.5.3 Záhloví a zápatí	21
1.5.4 Oddíly	22
1.6 Odkazy a popisky objektů	23
1.6.1 Křížové odkazy	23
1.6.2 Obrázky a tabulky	25
1.7 Dokumentové seznamy	26
1.7.1 Seznamy podle stylů	27
1.7.2 Rejstřík	28
1.7.3 Seznam literatury	29
1.8 Dlouhé dokumenty	33
1.9 Formáty souborů	34

2	Typografický systém T_EX	38
2.1	Jak to všechno funguje	38
2.1.1	Princip překládaných dokumentů	38
2.1.2	Jak vytvořit zdrojový soubor	41
2.2	Začínáme	42
2.3	Délkové jednotky	44
2.4	Písmo	44
2.4.1	Stupeň písma	44
2.4.2	Nastavení základních vlastností písma	45
2.4.3	Fonty	47
2.4.4	Zvláštní znaky a sekvence znaků	48
2.4.5	Diakritická znaménka	51
2.5	Odstavcové styly	52
2.5.1	Prostředí	52
2.5.2	Formát odstavce	52
2.5.3	Seznamy	54
2.6	Dokumenty	56
2.6.1	Třída dokumentu	56
2.6.2	Styly	56
2.6.3	Nadpisy	59
2.6.4	Titul a abstrakt	61
2.6.5	Poznámky pod čarou a na okraj	62
2.6.6	Záhlaví a zápatí	62
2.6.7	Dlouhé dokumenty	65
2.7	Vícesloupcová sazba	66
2.8	Křížové odkazy	69
2.9	Jak na chyby	70
2.10	Oživení textu	71
2.10.1	Vkládané obrázky	71
2.10.2	Nákresy	74
2.10.3	Tabulátory	78
2.10.4	Tabulky	79
2.10.5	Plovoucí prostředí	82
2.10.6	Boxy	84
2.10.7	Barvy	85
2.10.8	Hypertextové odkazy	86
2.11	Seznam literatury a rejstřík	88
2.12	Dokumentové seznamy podle stylů	89
2.13	Rozměry stránky	90
2.14	Píšeme pro vědu	92

2.14.1	Definice, věty a důkazy	93
2.14.2	Vzorce nejen matematické	94
2.14.3	Stromové struktury, diagramy a grafy	96
2.14.4	Postupy, algoritmy, předem zformátovaný text	98
3	OpenOffice.org	101
3.1	Kancelářský balík OpenOffice.org	101
3.2	Práce se styly	103
3.3	Písmo	104
3.3.1	Styly znaků	104
3.3.2	Zvláštní znaky	106
3.4	Odstavec	107
3.4.1	Odrážky a číslování	107
3.4.2	Styly odstavce	107
3.4.3	Iniciály	111
3.5	Struktura dokumentu	112
3.5.1	Styly stránky	112
3.5.2	Záhlaví a zápatí	113
3.5.3	Přechody mezi stránkami	114
3.6	Odkazy a popisky objektů	115
3.6.1	Křížové odkazy	115
3.6.2	Tabulky	115
3.6.3	Obrázky	115
3.6.4	Popisky objektů a odkazy	115
3.7	Dokumentové seznamy	115
3.8	Šablony	115
4	Co ještě přidám	116
Přílohy		
A	Užitečné klávesové zkratky	118
B	Další informace o $\text{\LaTeX}$	119
B.1	Nejdůležitější příkazy $\text{\LaTeX}$	119
B.2	Délkové registry a čítače v $\text{\TeX}$	119
B.3	Zajímavé třídy a balíčky	119
Závěr		120
Literatura		121
Rejstřík pojmů		126

Rejstřík příkazů a prostředí	131
Rejstřík balíčků, tříd a voleb	135

SEZNAM OBRÁZKŮ

1.1 Ukázka použití různých typů tabulátorů	4
1.2 Srovnání kurzívy a skloněného písma ve fontu Computer Modern . . .	7
1.3 Nastavení formátu odstavce	9
1.4 Víceúrovňové odrážky a číslování	11
1.5 Příklad hierarchie stylů	14
1.6 Nastavení číslování nadpisů	15
1.7 Vložení čísel stránek	20
1.8 Vložení poznámky pod čarou nebo vysvětlivky	21
1.9 Nastavení záhlaví a zápatí	22
1.10 Práce s oddíly	23
1.11 Vložení křížového odkazu na místo v textu	24
1.12 Vložení křížového odkazu na číslo nadpisu kapitoly	24
1.13 Vložení titulku tabulky a nastavení číslování obrázků	25
1.14 Automatický titulek objektu	26
1.15 Vložení obsahu do dokumentu	27
1.16 Vložení rejstříku do dokumentu	28
1.17 Citace s odkazy ve formě čísel	29
1.18 Citace s odkazy ve formě s příjmením autora	29
1.19 Číslování seznamu literatury a vytvoření odkazu na položku seznamu	30
1.20 Ukázka formátu záhlaví	32
1.21 Ukázka formátu obsahu	32
1.22 Vnořené dokumenty	33
1.23 Stránky Slunečnice.cz	36
2.1 Ukázka jednoduchého dokumentu	40
2.2 Prostředí programu T _E XnicCenter	41
2.3 Dialog pro vytvoření nadpisu a „záložky“, na kterou se lze odkazovat	42

2.4	Panel nástrojů s nástroji pro překlad dokumentu	43
2.5	Panel nástrojů pro změnu stupně písma	45
2.6	Panel nástrojů pro volbu řezu, tmavosti a rodiny písma	46
2.7	Určení sklonu čáry a vektoru	75
2.8	Slunečnice	83
2.9	Vykreslení rozměrů stránky příkazem <code>\layout</code>	92
3.1	Styly v OpenOffice.org	103
3.2	Nastavení efektů pro písmo ve stylu znaku	105
3.3	Nastavení víceúrovňového číslování	108
3.4	Nastavení toku textu odstavce	109
3.5	Nastavení číslování nadpisů	110
3.6	Nastavení iniciály pro odstavec	111
3.7	Ukázka použití iniciály pro první odstavec	112
3.8	Nastavení stylu stránky	113
3.9	Nástroj pro pokročilejší vkládání polí	114

SEZNAM TABULEK

2.1 Příkazy pro změnu stupně písma	45
2.2 Příkazy pro změnu řezu, tmavosti a rodiny písma	46
2.3 Nastavení parametrů odstavce	52
2.4 Možnosti zarovnání textu	53
2.5 Příkazy balíčku fancyhdr	64
2.6 Další příkazy balíčku graphics	72
2.7 Příkazy pro kreslení, které se píšou přímo do prostředí picture	76
2.8 Příkazy objektů vkládaných do parametrů příkazů <code>\put</code> a <code>\multiput</code> .	76
2.9 Příkazy pro rámečky v prostředí picture	77
2.10 Příkazy pro tabulátory	78
2.11 Ukázka tabulky s vlastními oddělovači	81

Úvod

Pár slov o typografii

Když na střední nebo vysoké škole, nebo třeba v zaměstnání, odevzdáváme dokument, jaký by měl být? Kromě samotného obsahu nás také zajímá forma – dokument by měl být přehledný, dobře čitelný, a v neposlední řadě také pěkný. Čím vyšší škola (vzděláním, nikoliv fyzicky), tím vyšší požadavky na množství, délku a kvalitu odevzdávaných dokumentů. Týká se to nejen vysokých škol technických a informačních, ale prakticky všech. Znalost vhodných postupů při psaní dokumentů se nám bude hodit třeba i při studiu ekonomie, jazyků nebo lesnictví.

Typografie je (nelekněte se) věda, která se zabývá správnou úpravou dokumentů. Měla by nás zajímat především tehdy, když vytváříme rozsáhlejší dokumenty o více stranách nebo desítkách či stovkách stran, ale znát typografická pravidla se vyplatí i u kratších dokumentů. V následujícím textu se budeme zabývat pouze základními *typografickými pravidly*, která nám budou stačit pro úpravu běžných dokumentů.

Pod pojmem *typografický systém* budeme chápat jakýkoliv systém (třeba program), který nám pomůže v dokumentu zachovávat typografická pravidla.

KAPITOLA 1

Typografie a MS Word

Program MS Word, součást kancelářského balíku MS Office, je vhodný především pro krátké texty, ale přesto je často používán i pro vytváření rozsáhlejších dokumentů. Samotný program příliš netlačí svého uživatele k zachovávání typografických pravidel a v některých případech dokonce jejich dodržování téměř znemožňuje, rozhodně není problém v něm vytvořit dokument porušující spoustu typografických zásad.

Přes tyto nedostatky se MS Word vyznačuje jednoduchým ovládáním, proto právě na něm si budeme postupně ukazovat, jak dodržovat nejdůležitější typografická pravidla. V této kapitole se nejdřív podíváme na možnosti úprav prostředí MS Wordu, abychom si dále dokázali zjednodušit práci, pak postupně probereme základní zásady, které je třeba dodržovat při psaní rozsáhlejších dokumentů, a jejich používání si ukážeme právě na MS Wordu.

Nakonec se naučíme zvládat tvorbu příliš dlouhých dokumentů, které (nejen) programu MS Word dělají problémy pro příliš velkou délku souborů, a také se podíváme na možnosti různých typů souborů, které vytváří samotný MS Word nebo programy, které s ním dokážou spolupracovat.

1.1 Uživatelská nastavení programu MS Word

Program MS Word je poměrně „uživatelsky přítulný“, tedy to, co nejčastěji potřebujeme provést, není těžké v prostředí programu najít. Často se však dostaneme do potíží nebo alespoň jsme zbytečně zpomalováni, pokud po programu chceme něco méně běžného, a to třeba opakovaně.

Týká se to například vkládání speciálních symbolů, používání dělení slov na konci řádků, označování pojmů, které se mají vložit do rejstříku, vkládání odkazů na vkládané objekty, kapitoly či stránky, atd. Standardní grafické rozhraní programu také trochu komplikuje používání případných vlastních stylů pro písmo.

Už při instalaci programu je dobré myslet na to, co budeme potřebovat. MS Word bohužel při instalaci používá jednu možnost nabízenou operačním systémem,

které většina uživatelů moc nerozumí až do chvíle, kdy na ni doplatí – *odložená instalace*.

Při instalaci si můžeme u každé součásti programu zvolit, zda ji přímo nainstalujeme nebo nenainstalujeme a nebo zda si přejeme instalovat později. Tato třetí možnost znamená, že vše bude vypadat přesně tak, jakoby součást byla nainstaloovaná, ale když ji chceme použít, objeví se pouze okno s informací, že se součást instaluje a je třeba vložit instalační CD. Když zrovna není po ruce, máme smůlu. Stává se to například s modulem pro dělení slov nebo kontrolu pravopisu.

Nyní se už budeme věnovat samotným úpravám. Používat hlavní menu, kontextová menu různých objektů na stránce a tlačítka na hlavním panelu (pod hlavním menu) umíme, stejně tak i lišty s měřítky. Na těchto lištách lze také nastavovat vlastnosti tabulátorů – po vytvoření tabulátoru (přímo na liště nebo v menu Formát) můžeme po poklepání na tento tabulátor určit vlastnosti tabulátorů – zda se k nim má text přichytit zleva, zprava nebo vycentrovat, nebo zda se má použít některý vodící znak.

Pokud klepneme pravým tlačítkem myši na hlavní panel (na volné místo, kde nejsou žádná tlačítka), zobrazí se nám seznam panelů – tam můžeme určit, které panely se zobrazí na hlavním panelu, vpravo či vlevo nebo na spodním okraji okna. Zobrazené panely můžeme libovolně přetahovat myší.

Pokud v tomto seznamu vybereme položku Vlastní, v následujícím dialogovém okně určujeme, které panely se zobrazí a co bude na jednotlivých panelech a v různých menu. Obsah kontextových menu určíme po vybrání položky Místní nabídka na první záložce okna. Tedy když některou akci používáme hodně často, přidáme její tlačítko na některý panel hlavního panelu. Tato vlastnost se také může hodit v případě, že v menu (hlavním či kontextovém) potřebujeme položku, která tam ve výchozím nastavení není.

Pro přehledné zobrazení struktury dlouhého dokumentu můžeme využít zobrazení osnovy (Zobrazit → Osnova) nebo rozložení dokumentu (Zobrazit → Rozložení dokumentu). Pro obě tyto akce existují malá tlačítka v levém dolním rohu hlavního okna programu. Tento režim je výhodný, pokud potřebujeme pracovat přímo se strukturou dokumentu – přesouvat celé kapitoly na jiná místa (kapitoly jednotlivých úrovní lze zbalit) nebo třeba měnit styl nadpisů či úroveň obsahu pro jednotlivé nadpisy.

Úlohy: _____

1. Zobrazte panel Formuláře. Přesuňte ho k pravému okraji (přichytte k okraji) a pak ho přesuňte nahoru na hlavní panel. Pak vypněte zobrazení tohoto formuláře.
2. Na hlavní panel umístěte tlačítka pro vložení symbolu, formát kapitálek, nastavení sloupců.
3. Pokud není zobrazen panel Tabulky a ohraničení, zobrazte ho. Přidejte na tento panel tlačítka pro přidání řádku a sloupce.
4. Vytvořte dokument s alespoň dvěma úrovněmi nadpisů a běžným textem. Vyzkoušejte zobrazení osnovy a rozložení dokumentu (na dokumentu, ve kterém

jsou použity styly nadpisů). Na hlavním panelu najdete panel pro práci s dokumentem v režimu osnovy (jsou na něm kromě jiného různé šipky a čísla) a vyzkoušejte tlačítka na tomto panelu (textový kurzor přesuňte na některý nadpis).

5. Pouze pomocí tabulátorů vytvořte podle obrázku 1.1 na straně 4 úsek zprávy ze služební cesty polárního badatele Jana Nováka.

Počet	sobů	126	
	lišek	12	
	zajíců	54	
Celkem zvířat		192	
Diety			4 500,--
- z toho	ubytování	2 000,--	
	snídaně	480,--	
	obědy	1 500,--	
	večeře	520,--	
<u>Poplatek za vstup do rezervace</u>			<u>5 924,--</u>
Finanční náklady celkem			10 424,--
V Laponsku dne 2. 11. 2006			
			
			Jan Novák

Obrázek 1.1: Ukázka použití různých typů tabulátorů

Tabulátory můžete tvořit buď klepnutím pravým tlačítkem myši na pravítko nad textem nebo v hlavním menu Formát → Tabulátory. Některé jsou zarovnané doleva, jiné doprava, nadpis a tečky na střed (poklepejte na pravítko na některý tabulátor).

1.2 Písmo

V této části kapitoly budeme používat především dialogové okno Písmo, ke kterému se dostaneme v menu Formát → Písmo, Automatické opravy z menu Nástroje → Možnosti automatických oprav a Symbol z menu Vložit → Symbol.

1.2.1 Nastavení písma

Pozorný uživatel programu si zřejmě všiml, že má na výběr mezi více různými *typy písem (fonty)* – Times New Roman, Arial, Symbol, Book Antiqua, ... Tato písma dělíme podle různých kritérií a určíme jejich vhodnost použití pro daný účel.

Písma můžeme dělit na *proporcionální* a *neporcionální*. U neporcionálního písma zabírají všechna písmena vždy stejnou šířku, tato písma jsou charakteristická pro staré mechanické psací stroje. V nabídce programu MS Word je to písmo Courier New. Neporcionální písma nejsou vhodná pro běžný text v dokumentech, používají se pouze pro speciální účely, kdy je vhodné mít písmena „zarovnaná do sloupců“ a navzájem dobře oddělená, například ve výpisech programového kódu nebo u WWW adres.

Naopak u proporcionálních písem zabírají různá písmena různou šířku, například písmeno i potřebuje méně místa než písmeno m. V dalším textu se budeme zabývat spíše písmy proporcionálními.

Písma dále dělíme na *patková* (serifová) a *bezpatková* (grotesková). Patky (serify) jsou vodorovná zakončení tahů písmen, a jak se můžeme sami přesvědčit, v nabídce MS Wordu je najdeme například u písma Times New Roman, Book Antiqua, Palatino, Garamond, atd. K písmům bezpatkovým patří třeba Arial Lucida, a také MS Sans Serif, třebaže má v názvu slovo „serif“ (slovo *sans* znamená ve francouzštině bez, jde tedy o písmo „bez serifů“).

Patková písma jsou obecně čitelnější, proto se hodí i pro rozsáhlé dokumenty, zatímco bezpatková se doporučují spíše pro velmi krátké a dobře členěné texty, jako jsou například prezentace, reklamní materiály, nápisy, titulní strany tiskovin. Patková a bezpatková písma kombinujeme jen výjimečně na téže stránce, nanejvýš je možno je použít pro nadpisy (běžný text by měl být psán patkovým písmem), přesto je lepší použít zvýraznění nadpisů jiného typu (třeba tučné písmo).

Samostatnou skupinu tvoří písma *kaligrafická* (písmomalířská, na počítači napodobující podobu psaného písma), zdobená a speciální (jednotlivá písmena jsou vlastně znaky písmo nepřipomínající). Nejsou vhodná pro běžné dokumenty, používají se jako doplněk obvykle u reklamních tiskovin.

U písma nás tedy zajímá jeho typ a vlastnosti daného typu, jak bylo probráno výše. Dále nás zajímá *řez písma*, což je například kurzíva, tučné písmo, stínované, skloněné (slanted), apod.

Kuželka je oblast, do které se vejde kterékoliv písmeno daného písma (od „linky“ nahoru). Její výšku nazýváme *stupeň písma* (někde velikost), je to tedy výška písma. Udává se v *typografických bodech* a obvyklé hodnoty jsou 8, 10, 12, ... bodů.

Typ písma, jeho řez a další vlastnosti znaků nastavujeme v dialogovém okně Písmo. Dialogové okno má několik záložek, z nichž první slouží k nastavení typu a řezu písma a například jeho barvy. Druhou záložku, Proložení znaků, využijeme, pokud chceme upravit mezery mezi znaky (není moc vhodné) nebo znaky posouvat vertikálně.

1.2.2 Zvláštní znaky

S příchodem proporcionálního písma vznikla postupně typografická pravidla, která odlišovala různá použití podobných znaků. My se zde podíváme pouze na základní případy, které se v dokumentech běžně objevují.

Spojovník najdeme přímo na klávesnici, neobklopuje se mezerami. Slouží nejen k rozdělení slova na konci řádku, ale také k připojení podmínkové části slova

„-li“ (bude-li, vidí-li) a ke spojení více slov názvu jediného kompaktního objektu (česko-anglický slovník, Ostrava-jih).

Pomlčka je delší než spojovník, na klávesnici ji nenajdeme. V MS Wordu bývá generována automaticky po dopsání následujícího slova, nebo lze určit, aby se místo dvojice spojovníků -- sázela pomlčka (dialogové okno Automatické opravy, záložka Automatické úpravy formátu při psaní). Lze také použít dialogové okno Symbol, záložku Speciální znaky (krátká pomlčka), nebo klávesy  +  ¹.

Pomlčka se používá buď jako náhrada čárky v souvětí, v tom případě se obklopuje mezerami (napsala – pro upřesnění –), a nebo pro vyjádření rozsahu, a pak se mezerami neobklopuje (10–15 minut, zápas Sparta–Slavia, trasa Praha–Prčice). Druhý případ poznáme také podle toho, že pomlčku zde můžeme nahradit některým slovem (například až, proti, z . . . do, mezi . . . a).

*Nezlomitelná mezer*a je mezer, která nesmí být na konci řádku. Používá se především za jednoznakovými předložkami a spojkami (k, z, v, o, a, i, . . .), výjimkou je spojka „a“ v úzkém sloupci. V MS Wordu lze místo ní vložit *pevnou mezeru*.

*Pevná (zúžená) mezer*a je nezlomitelná mezer, která má pevnou délku (nesmí být rozšířena). V MS Wordu ji vytvoříme stisknutím kláves  +  +  nebo na druhé záložce v dialogovém okně Symbol (pevná mezer), v některých případech ji Word vloží sám. Používá se:

- pro rozčlenění dlouhého čísla, například 12 004 891 nebo oddělení čísla a jednotky (15 kg),
- ve jméně pro oddělení iniciály od příjmení (J. K. Tyl).

Tečka – před tečkou neděláme mezeru, za ní ano, mezi tečku a pravou závorku však mezeru nedáváme a totéž platí i pro uvozovky.

Trojtečka neboli *výpustka* je tvořena třemi tečkami . . . , které nejsou odděleny mezerami (ve skutečnosti je mezi nimi zúžená mezer), za trojtečku už další tečku neděláme, i kdyby šlo o konec věty. V programu MS Word je obvykle sázena automaticky po napsání tří teček za sebou, ale můžeme ji vložit z dialogového okna Symbol nebo použít klávesovou zkratku  + .

Symbol krát (×) nikdy nenahrazujeme písmenem x („iks“)! Tento symbol najdeme v dialogovém okně Symbol, pak píšeme třeba 3× (bez mezery).

Uvozovky v českém textu odlišujeme vstupní (levé) a výstupní (pravé). Vstupní se píšou dole a mají tvar devítek, výstupní se píšou nahoře a mají tvar šestek. MS Word obvykle uvozovky takto formátuje automaticky (lze nastavit v dialogu Automatické opravy, záložka Automatické úpravy formátu při psaní).

Kromě těchto „oblych uvozovek“ lze použít ještě »úhlové uvozovky«. V české typografii úhlové uvozovky míří k uvozovanému textu, zatímco ve francouzštině, kde se hodně používají, míří obráceně – směrem od uvozovaného textu.

¹  znamená znak „-“ na numerické části klávesnice vpravo.

Promíle a procento se od čísla odděluje mezerou (10 %, 2 ‰). Pokud mezeru nenapišeme, jde o přídavné jméno „desetiprocentní“, které se však většinou vypisuje slovně.

Stupeň se od čísla odděluje mezerou, pokud za stupněm má následovat Celsius, jinak mezeru nepíšeme (25 °C, 45°).

Slitky (ligatury) jsou shluky několika písmen, která k sobě opticky patří a lze je spojit do jednoho „multiznaku“. Typický příklad je dvojice písmen fi, fl, ff. MS Word se slitky sám nezabývá, ale některé můžeme najít v dialogu Symbol (vždy hledáme v tom typu písma, ve kterém právě pracujeme).

Podřezávání (kerning) je přisouvání znaků, aby zbytečně nevznikala příliš velká mezera mezi písmeny. Používá se u dvojic znaků, které se svými přisunutými okraji tvarově doplňují, například AV nebo Te. Ve Wordu si nanejvýš můžeme pohnout s přisouváním znaků na záložce Proložení znaků v dialogu Písmo.

Minusky, verzálky, kapitálky – minusky jsou malá písmena, verzálky jsou VELKÁ PÍSMENA, kapitálky jsou „zmenšená“ velká písmena – PÍSMENA, mohou být použity například pro nadpisy nejvyšší úrovně nebo příjmení.

1.2.3 Vyznačování

V odborném textu potřebujeme také vyznačovat (zvýrazňovat). Máme na výběr mezi těmito možnostmi:

1. *Kurzíva* (italika) se používá pro běžné vyznačování. Její výhodou je, že sice plní svou roli (zvýrazňuje vyznačený text), ale zároveň nenarušuje barevnou koncepci stránky.

Pokud potřebujeme vyznačit něco, co se nachází v textu psaném kurzívou, nastavíme normální písmo, abychom zachovali vlastnosti z předchozího bodu.

2. *Skloněné písmo* (slanted, v MS Wordu není) používáme pro vyznačení v textu málokdy, například tehdy, když chceme přepsat něco, co i jinde je psáno tímto písmem, nebo odlišit vyznačení od kurzívy. Skloněné písmo je trochu jiné než kurzíva, viz obrázek 1.2 (v některých písmech to však moc neplatí, například v zde použitém).

Kurzíva: <i>Text psaný písmem Computer Modern</i> Skloněné: <i>Text psaný písmem Computer Modern</i>
--

Obrázek 1.2: Srovnání kurzívy a skloněného písma ve fontu Computer Modern

3. **Tučné písmo** (nebo polotučné pro některé typy písma) se používá spíše tam, kde cíleně chceme přitáhnout pozornost čtenáře, například pro klíčová slova ve slovnících, v některých učebnicích, v novinách, pro nadpisy. V textu psaném tučným písmem zvýrazňujeme **tučnou kurzívou**.

4. KAPITÁLKAMI se vyznačují jména osob (u citátů nebo v seznamu literatury).
5. Neproporcionálním písmem se vyznačuje výpis zdrojového kódu nebo texty vypisované jako výstupy programů, uvnitř textu psaného neproporcionálním písmem můžeme navíc zvolit *kurzívu* nebo **tučné písmo**. Pokud potřebujeme ještě navíc odlišit výpis zdrojového kódu od výstupu programu, můžeme jedno z toho například dát do rámečku.
6. Bezpatkové písmo lze použít pro vyznačení v textu psaném v písmu patkovém, pokud má alespoň trochu podobnou kresbu (písmena vypadají podobně, až na patky a třeba velikost a mírný sklon). V typografických systémech máme obvykle alespoň pro některé typy písem variantu s patkami a bez patek, v MS Wordu ale tomu tak není. Pro použití bezpatkového písma bychom měli mít i další důvod než vzhledový, zde je to například používání bezpatkových písem pro nápisy ve výchozím nastavení MS Wordu a dalších programů ve Windows.
7. Velice efektním způsobem vyznačení je ikonka (malý obrázek) umístěná například u okraje. Obvykle tak bývají vyznačeny části textu nebo odstavce se speciálním významem – úlohy, definice nového pojmu, poznámka, apod.

 Vyznačující ikonka nemusí být u okraje, může to být obrázek na začátku řádku obtékaný textem (takový obrázek by neměl být moc velký, ideální výška je na dva řádky). Význam a použití jsou stejné jako u bodu 7 předchozího výčtu.

Než začneme psát dokument, měli bychom si ujasnit, co budeme jak vyznačovat. Můžeme použít více různých způsobů vyznačení, ale tak, aby celek byl harmonický, a nepřeháníme to. Je vhodné v kapitole s předmluvou nebo úvodem čtenáře se způsobem vyznačování seznámit.

Úlohy: _____

1. *Procvičení výběru*: Spustte program MS Word a napište jakýkoliv odstavec delší než jeden řádek. Teď myší poklepejte (t. j. dvakrát klepněte) na některé slovo v odstavci. Zrušte výběr (klepněte jinam) a třikrát rychle za sebou klepněte na některé slovo odstavce. Jaký je rozdíl?

Ještě si procvičíme výběr řádků. když myší najedeme úplně doleva před text (pořád zůstáváme na plátně, kurzor je ve tvaru šipky) a klepneme myší, vybereme jeden řádek. Pokud místo klepnutí jen stiskneme (nepustíme) tlačítko myši a „jedeme“ dolů či nahoru, vybereme více řádků – výběr skončí na tom místě, kde tlačítko pustíme.

2. Projděte si dialogová okna Písmo, Automatické opravy (zvláště první část záložky Automatické úpravy formátu při psaní) a Symbol.
3. Zjistěte:
 - jak do textu vložit znak © a jak napsat úhlové uvozovky,
 - jak pomocí klávesové zkratky vložit do textu slitek (ligaturu) fi,

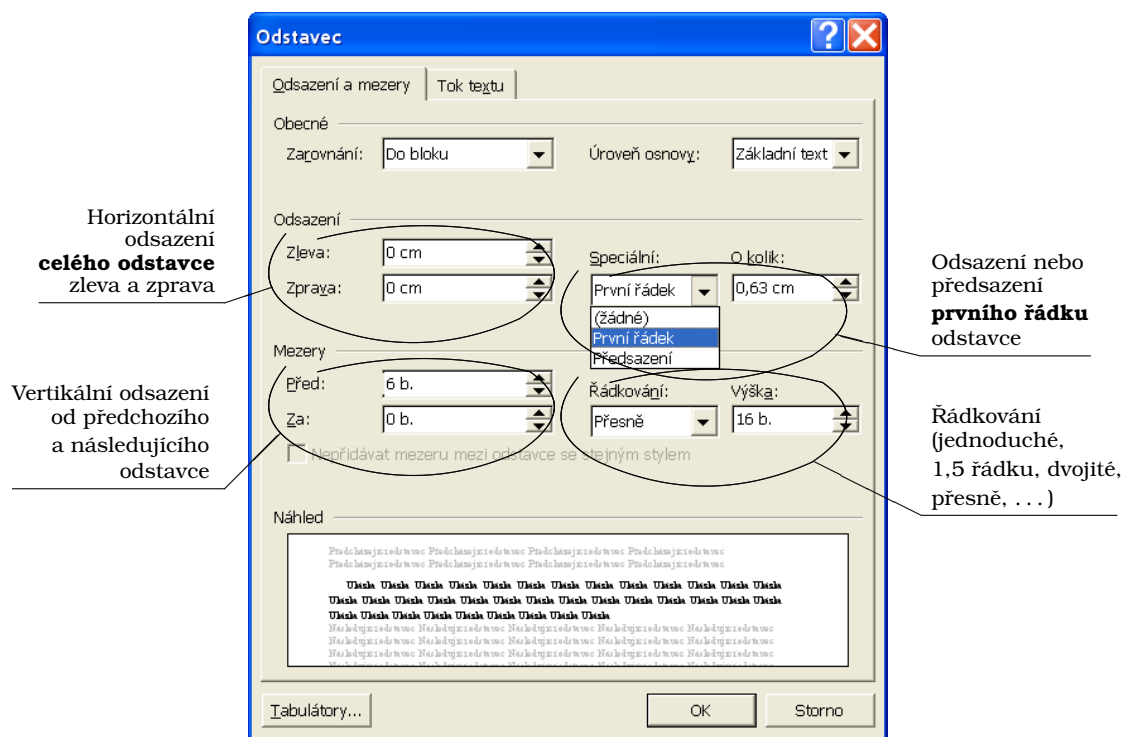
- jak tuto klávesovou zkratku změnit na **Ctrl** + **F** + **I** (v určitém dialogovém okně přesunete kurzor na jednu jeho část a pak přímo danou zkratku použijete),
- vyzkoušejte si změnu minusek a verzálek na kapitálky, porovnejte malá písmena kapitálek s verzálkami.

1.3 Odstavcové styly

V této části kapitoly budeme používat především dialogová okna Odstavec z menu Formát → Odstavec, Odrážky a číslování z menu Formát → Odrážky a číslování (v něm tlačítko Vlastní) a případně další položky z menu Formát.

1.3.1 Formát odstavce

Text by měl být členěn do odstavců, a to tak, že jeden odstavec představuje jednu ucelenou myšlenku, rozhodně bychom neměli psát stylem „co kapitola, to odstavec“.



Obrázek 1.3: Nastavení formátu odstavce

Odstavce musí být opticky odděleny, a to buď vertikálním odsazením od předchozího odstavce nebo nastavením odsazení prvního řádku² (nebo obojím). Levý

²Na psacím stroji se první řádek odsazoval pěti údery na mezník. Tento postup na počítači nepoužíváme!

a pravý okraj také můžeme odsadit, čímž vznikne zúžený sloupec textu, to se používá především pro dlouhé citace. Všechny tyto úpravy provádíme v dialogovém okně Odstavec.

Při *odsazení prvního řádku* u běžného odstavce se doporučuje použít hodnotu v rozmezí 1–2 čtverčíky, pod pojmem *čtverčík* rozumíme délku kuželky (viz str. 5). V tomto dokumentu je použita horní hranice tohoto rozmezí. V programu MS Word se bohužel používá pro tento účel metrická míra, zde můžeme volit jako horní hranici 0,63 cm. Přednastavená hodnota 1,25 je příliš velká, použijeme ji nanejvýš při spíše větším písmu. V prvním odstavci kapitoly se první řádek zpravidla neodsazuje.

Pokud se rozhodneme *odsadit odstavce vertikálně* (tj. od předchozího a následujícího odstavce), mezeru volíme tak, aby byla viditelná, ale ne příliš velká. Ve většině typografických systémů se určuje, kolik přidáme oproti vzdálenosti řádků uvnitř odstavce, tedy jednoduše zadáme nenulovou hodnotu (obvykle 3 body pro menší písmo, 6 bodů pro větší písmo).

Pro lepší čitelnost složitěho textu můžeme zvětšit *řádkování*, doporučuje se přidat maximálně polovinu stupně písma. Vzdálenost mezi spodními okraji řádků uvnitř odstavce se nastavuje asi na 120 % stupně písma (viz str. 5). Standardní nastavení v MS Wordu toto pravidlo celkem respektuje, proti typografickým pravidlům je však „volnější chování“ v případě, že je pro některé slovo zvolen vyšší stupeň než pro zbytek odstavce nebo do textu vložíme objekt větší než je stupeň písma – pak tento řádek bude od předchozího více oddálen. Nastavení můžeme změnit v dialogu Odstavec, kde řádkování změníme z *Jednoduché* na *Přesně* danou hodnotu.

Stránkový zlom (tedy konec stránky) by správně neměl oddělit jediný řádek delšího odstavce, ať už na začátku nebo na konci odstavce. Způsob vkládání stránkového zlomu se v MS Wordu řídí v dialogu Odstavec, záložka Tok textu.

1.3.2 Seznamy

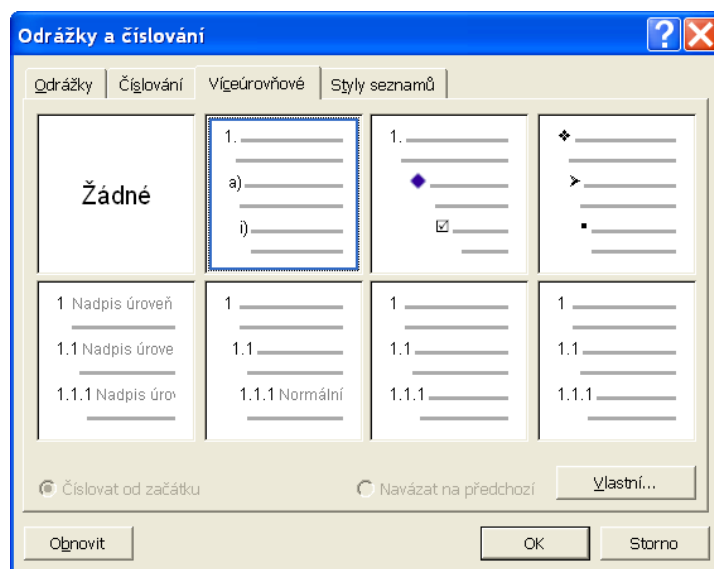
Také seznamy jsou odstavce. Rozlišujeme tři základní druhy seznamů, které můžeme libovolně vnořovat:

- výčtové seznamy, položky jsou vyznačeny vlevo odrážkami,
- číslované seznamy s čísly nebo písmeny a oddělovacím znakem (tečka, závorka),
- seznamy s popisky (obdobně formátované jako položky slovníku, viz například seznam na str. 5).

Navzájem vnořené seznamy nazýváme *víceúrovňové*, nastavujeme je v okně Odrážky a číslovaní na záložce Víceúrovňové (viz obr. 1.4 na str. 11), upravujeme je v dalším ziskáném přes tlačítko Vlastní.

Pokud chceme přímo v seznamu změnit úroveň odrážky, se kterou právě pracujeme, přeneseme textový kurzor na začátek položky a použijeme klávesu  pro následující úroveň nebo ³ pro předchozí úroveň číslovaní. V textu příslušejícímu jedné odrážce (příp. číslu) můžeme mít i více než jen jeden odstavec – když chceme pokračovat novým odstavcem uvnitř téže odrážky, místo klávesy  použijeme na konci odstavce klávesovou zkratku  + .

³Na klávesnici může být klávesa  označena .



Obrázek 1.4: Víceúrovňové odrážky a číslování

Seznamy lze zleva odsadit, ale ne příliš (půl stránky je opravdu moc), položky seznamu bývají zpravidla navzájem vertikálně odsazeny podobně jako odstavce (kromě seznamů s velmi krátkými položkami). Různé druhy seznamů můžeme navzájem vnořovat, ale platí zásada, že „podřízený“ seznam by měl mít jiné značení než seznam o úroveň výše. První dva druhy seznamů nastavujeme v dialogovém okně Odrážky a číslování, a to včetně nastavení vlastního značení, což se hodí zvláště u víceúrovňových seznamů.

Číslované seznamy se v odborných publikacích (včetně diplomových prací) používají také při definování stylu nadpisů (podrobněji v kapitole 1.4).

Vytvoření seznamů s popisky je v MS Wordu trochu náročnější. Jde vlastně o speciální typ odstavce, který má název, za ním není symbol přechodu na nový řádek (text odstavce pokračuje přímo za názvem), a první řádek je textu odstavce předsazen (ne odsazen, jak je to obvyklé u běžných odstavců). Nouzově můžeme tento odstavcový styl vytvořit tak, že nastavíme předsazení (ne odsazení!) prvního řádku a spolehneme se na to, že při používání tohoto seznamu nezapomeneme vždy popisek zvýraznit třeba kurzívou nebo tučným písmem.

Úlohy: _____

1. Všimněte si, jak jsou v seznamu na str. 10 použita *interpunkční znaménka* (tečky a čárky ve větách), zvláště na konci druhé odrážky.
2. Vytvořte jakýkoliv odstavec s alespoň dvěma řádky a nastavte jeho formát tak, aby první řádek (pouze první řádek) byl odsazen o 0,63 cm, před odstavcem bylo vertikální odsazení od předchozího odstavce 6 bodů a vzdálenost mezi řádky byla *přesně* 16 bodů (tedy nastavujete řádkování). Odstavec zarovnejte do bloku.

3. Pro některý odstavec vytvořte formát vhodný pro citace na začátku knihy nebo kapitoly: zmenšete písmo o 1 bod, nastavte odsazení celého odstavce zleva i zprava o 2 cm, nastavte kurzívu a zarovnejte do bloku.
4. Vytvořte víceúrovňový seznam s alespoň dvěma úrovněmi, kde první je číslovaná a druhá s odrážkami (jde tedy o víceúrovňový seznam). Číslování jsou zarovnaná vpravo, a to na 0,5 cm, za číslem bude následovat tečka. V druhé úrovni zvolte místo číslování některé odrážky, samotnou odrážku zarovnejte na 1 cm. V obou úrovních zajistěte, byly všechny řádky včetně prvního zleva zarovnané (tj. v umístění textu bude vhodné odsazení).
5. V druhé úrovni seznamu z předchozího příkladu nastavte jako odrážku některý ze symbolů z dialogového okna Symbol (provedte tam, kde jste pro druhou úroveň nastavovali odrážky místo číslování). Vyzkoušejte si klávesové zkratky z kapitoly 1.3.2.

1.4 Styly a šablony

V této kapitole budeme používat dialogová okna Zobrazit formátování a Styly a formátování, obě položky v menu Formát, a dále Šablony a doplňky v menu Nástroje → Šablony a doplňky.

Pokud chceme v celém (dlouhém) dokumentu zachovat stejný formát, měli bychom používat styly. *Styl* je souhrn informací o formátování odstavců, nadpisů a dalších objektů v dokumentu. Práce se styly je důležitá nejen pro udržení formátu v dokumentu, ale také pro jeho snadnou a rychlou změnu a snadnější přenositelnost na jiné dokumenty.

Základní údaje o nastaveném stylu zjistíme v dialogu Zobrazit formátování, vlastní nastavení můžeme provést buď přímo zde nebo v dialogu Styly a formátování, druhý dialog je pro provádění některých nastavení vhodnější.

V MS Wordu jsou styly ukládány do souborů s příponou .dot, kterým říkáme *šablony*. K dispozici máme základní styly definované v šabloně normal.dot.

1.4.1 Vhodné nastavení stylů

Předdefinované styly bohužel nemůžeme považovat za vhodné pro delší dokumenty. Doporučuje se předefinovat téměř vše, především styl nadpisů⁴. Pro nadpisy platí, že jejich typ písma by měl odpovídat typu písma běžného textu (alespoň opticky), nadpis nesmí být posledním řádkem na stránce a měl by být od předchozího textu odsazen více než od následujícího. Nadpisy první úrovně začínají vždy na nové stránce.

⁴Dokonce ani typ písma Times New Roman není příliš vhodný, i když jde o patkové písmo, a to z estetických důvodů. Toto písmo bylo navrženo spíše pro noviny, kde je nutné text co nejvíc „natěsnat“, i na úkor příjemného vzhledu písma.

V delších dokumentech odborného rázu⁵ jsou kapitoly členěny do *úrovní* (obvykle tři úrovně, více úrovní zhoršuje přehlednost). Nadpisy různých úrovní by měly být opticky rozlišeny, tedy pro různé úrovně postupně nastavíme buď různé řezy (kapitálky, normální, kurzíva) nebo stupeň s případným jiným vyznačením (například pro text velikosti 12 bodů volíme pro nadpisy stupeň 16, 14 a 13 bodů, všechny úrovně tučným písmem).

Nadpisy členěné do úrovní by rozhodně měly být *číslovány*. Používáme víceúrovňové číslování a pro zvýšení přehlednosti musíme zajistit, aby u nadpisů nižší úrovně bylo také číslo nadřazených úrovní, obvykle se čísluje čísly arabskými (1, 2, 3, 4, ...). Tečka se zde narozdíl od běžných seznamů dělá jen v sekvenci čísel mezi čísly různých úrovní, ne na konci sekvence (například pro desátou podkapitolu druhé kapitoly je to 2.10 bez tečky za číslem 10, jak je vidět i na tomto dokumentu).

Styly můžeme načíst z existující šablony (standardně jsou načteny ze šablony `normal.dot`, jinou volíme v dialogu Šablony a doplňky) a případně pozměnit v dialogu Styly a formátování, předdefinované styly se pak ukládají přímo do dokumentu.

Nic nám však nebrání ve vytvoření vlastní šablony, kterou pak můžeme používat i pro další dokumenty. To jednoduše provedeme tak, že v (prázdném) dokumentu nastavíme styly podle vlastního uvážení a dokument pak uložíme s příponou `.dot` (Soubor → Uložit jako, typ souboru zvolíme Šablona dokumentu). V dokumentu, kde chceme naši šablonu použít, ji načteme v dialogu Šablony a doplňky, tlačítko Připojit. Pokud chceme, aby se každá změna na šabloně projevila i v našem dokumentu, zaškrtneme volbu Automaticky aktualizovat styly dokumentu. Tlačítko Organizátor nám umožní zjistit podrobnější informace o načtené šabloně.

1.4.2 Postup vytvoření hierarchie stylů

Nyní se podíváme na správný postup definování stylů. Máme dvě možnosti – buď pozměníme existující styly nebo na jejich základě vytvoříme nové. Obvykle stačí jen upravit existující styly, někdy se však nevyhneme vytváření nových. Zde si ukážeme vytvoření vlastních stylů, na úpravu stylů se podíváme v úlohách.

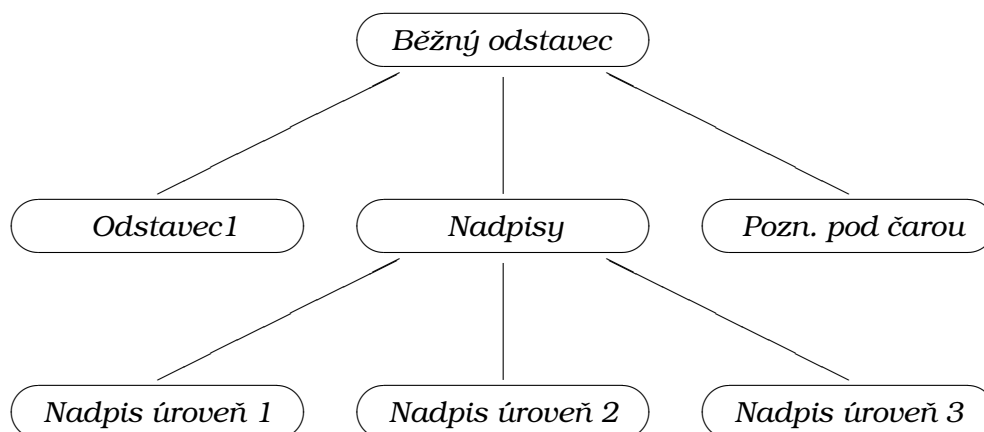
Abychom si zjednodušili práci pro případ, že některé vlastnosti stylů budeme měnit, uspořádáme je do hierarchie (viz například obr. 1.5). Každý styl definujeme z některého existujícího, přesněji z toho, který je v hierarchii nad ním. Tím umožníme „dědění“ základních vlastností, jako je třeba typ použitého písma.

Ne všechny styly nutně musí být použitelné v dokumentech. Můžeme vytvořit i pomocný styl, který bude obsahovat vlastnosti společné více různým stylům, jeho potomkům v hierarchii (podle obr. 1.5 například styl *Nadpisy*). Tyto potomky již budeme přímo používat.

Vytvoření hierarchické struktury z obrázku 1.5 si ukážeme na souhrnném příkladu.

1. Zobrazíme dialogové okno Styly a formátování a najdeme styl *Normální*. Na něm založíme náš styl běžný text pojmenovaný *Běžný odstavec*, kterým budeme začínat. Klepneme na tlačítko *Nový styl* a v dialogovém okně pro definování

⁵Dokumenty odborného rázu jsou všechno kromě beletrie.

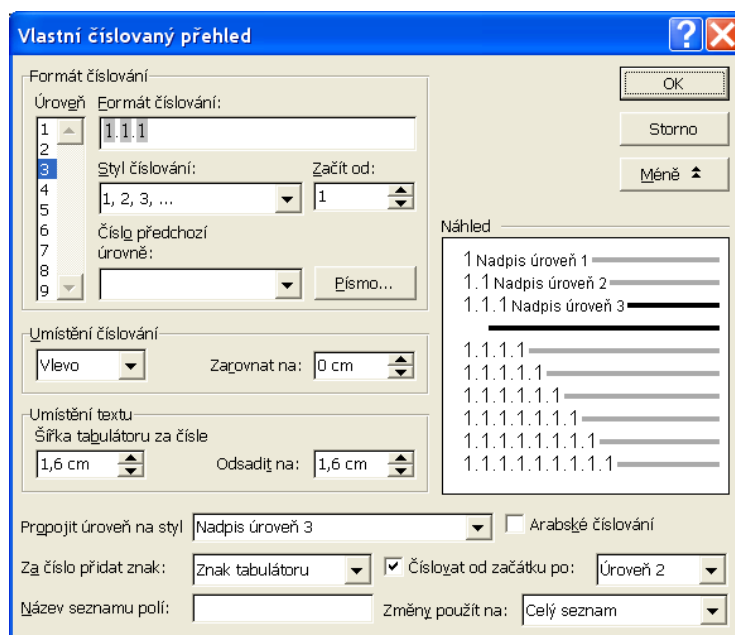


Obrázek 1.5: Příklad hierarchie stylů

nového stylu určíme všechny potřebné vlastnosti, využijeme také tlačítko Formát vlevo dole. Pro styl *Běžný odstavec* provedeme tyto změny:

- styl nejdřív pojmenujeme (*Běžný odstavec*), abychom se dále mohli na tento název odkazovat,
 - styl následujícího odstavce nastavíme na tento styl,
 - nastavíme písmo *Palatino Linotype*,
 - text zarovnáme do bloku, velikost nastavíme na 11 bodů,
 - první řádek odstavce odsadíme o 0,63 cm,
 - vertikální mezeru před odstavcem nastavíme na 3 body.
2. Potomky v hierarchii budeme definovat tak, že vytvoříme nový styl a při určování vlastností nového stylu vybereme v seznamu Styl založený na daného předka (předka najdeme v seznamu Styl založený na).
- Ze stylu *Běžný odstavec* odvodíme *Odstavec1*, kde odebereme vlastnost odsazení prvního řádku (tento styl použijeme vždy pro první odstavec kapitoly), styl následujícího odstavce nastavíme na *Běžný odstavec* a ještě zkontrolujeme, zda je nastaven *Běžný odstavec* jako předek.
3. Teď vytvoříme styl *Nadpisy* shrnující vlastnosti všech nadpisů. Provedeme tato nastavení:
- po pojmenování styl založíme na *Běžný odstavec*,
 - styl následujícího odstavce bude *Odstavec1*,
 - nastavíme tučné písmo a zarovnání textu doleva,
 - zrušíme odsazení prvního řádku, vertikální mezeru před odstavcem nastavíme na 18 bodů a za ním na 12 bodů,
 - na záložce Tok textu zaškrtneme políčka Svázat řádky a Svázat s následujícím.
4. Na stylu *Nadpisy* založíme styl *Nadpis úroveň 1*:
- nastavíme velikost písma 16,

- pro následující odstavec určíme *Odstavec1*,
- přes tlačítko Formát nastavíme číslování – zvolíme víceúrovňové číslování a přes tlačítko Vlastní provedeme další úpravy, především smazání tečky za posledním číslem (pokud tam je) a ve spodní části dialogu vybereme propojení první úrovně na tento styl, ukázka dialogového okna s nastavením pro třetí úroveň je na obrázku 1.6 na str. 15,
- u vlastností odstavce zaškrtneme na záložce Tok textu vlastnost Vložit konec stránky před, aby hlavní kapitola začínala vždy na nové stránce.



Obrázek 1.6: Nastavení číslování nadpisů

- Podobně vytvoříme styly *Nadpis úroveň 2* a *Nadpis úroveň 3*. Předkem bude styl *Nadpisy*, velikosti písma zvolíme 14 a 13, pro následující odstavec opět určíme styl *Odstavec1* a u číslování u druhé a třetí úrovně zvolíme tyto styly.
- Nyní vytvoříme styl pro poznámku. Můžeme zde upravit existující styl *Text poznámky pod čarou*, který najdeme v seznamu stylů v okně *Styl a formátování* (po zvolení zobrazení všech stylů).

Pokud se nám nedaří tento styl v seznamu najít, můžeme hledání obejít tak, že do dokumentu (nebo šablony) vložíme poznámku pod čarou (menu *Vložit* → *Odkaz* → *Poznámka pod čarou*), při vkládání zároveň určíme některé vlastnosti tohoto stylu – poznámky budou umístěny na konec stránky, formát číslování nastavíme na velká římská čísla (I, II, III, ...) a číslování nebude průběžné, ale pro každou stránku zvlášť⁶. Po vložení poznámky ji najdeme na konci stránky a vyvoláme kontextové menu⁷ textu poznámky (i když tam žádný není –

⁶Číslování poznámek pro každou stránku zvlášť je vhodné, pokud je v dokumentu těchto poznámek hodně.

⁷Kontextové menu objektu vyvoláme stisknutím pravého tlačítka myši na tomto objektu.

nastavíme kurzor za značku poznámky) a vybereme položku Styl, v zobrazeném okně bychom měli mít v seznamu vybranou položku *Text poznámky pod čarou*.

Stiskneme tlačítko Změnit, změníme název na Poznámka pod čarou, nastavíme předka na *Běžný odstavec*, velikost písma zmenšíme o 2 (na 9 bodů) a zaškrtneme přidání do šablony a automatickou aktualizaci.

V dialogovém okně pro vytvoření (ale také úpravu) stylu) jsou zaškrťovací políčka Přidat do šablony a Automaticky aktualizovat. Pokud zaškrtneme první z nich, pak jakékoliv změny na definici stylu, které provedeme, se projeví i v šabloně, odkud tento styl pochází (včetně `normal.dot`). Tuto možnost používáme pouze pro styly z vlastních šablon, u kterých opravdu chceme změny provádět.

Možnost automatické aktualizace zase znamená, že když provedeme „manuální“ změnu formátování v dokumentu na textu, u kterého je nastavený tento styl, změna, kterou provedeme, se projeví i na definici stylu samotného. Například když řez textu ve stylu *Nadpis 1* změníme na kapitálky bez tučného písma, tato změna se objeví i v daném stylu a projeví se na všem, pro co jsme tento styl použili.

V dialogovém okně z tlačítka Nový styl můžeme v MS Wordu od verze XP také vytvořit styly pro tabulky a seznamy, stačí zvolit v rozbalovacím seznamu Styl založený na danou položku. Tyto styly se tvoří obdobně, proto se jimi zde nebudeme zabývat.

Při psaní dokumentu vybíráme styly přímo v horní liště okna MS Wordu. Tam se samozřejmě „pletou“ také styly předdefinované. Pokud chceme mít rychlejší přístup ke svým vlastním stylům, dá se zařídit, aby se nacházely v horní části seznamu stylů. To můžeme provést třeba tak, že prvním znakem názvu stylu bude znak, který se řadí před písmena abecedy, například *podtržítko*.

Úlohy: _____

1. Otevřete nový dokument a nastavte v něm styly podle souhrnného příkladu v této kapitole. Po dokončení soubor uložte jako šablonu (s příponou `.dot`) do složky, která vám bude nabídnuta.
 2. Otevřete nový dokument a načtěte do něho šablonu, kterou jste vytvořili v předchozím bodě. Nezapomeňte zaškrtnout políčko pro automatickou aktualizaci stylů dokumentu.
 3. U předdefinovaných stylů proveďte následující změny:
 - u stylu *Normální* nastavte typ písma na *Bookman Old Style*, zarovnání do bloku a vertikální odsazení od následujícího odstavce na 6 bodů,
 - u stylu *Nadpis 1* zvolte číslování s tím, že vytvoříte nový styl víceúrovňového číslování, kde položky úrovně budou vázány na styly *Nadpis 1*, *Nadpis 2* a *Nadpis 3* (pokud takové číslování ještě neexistuje), pokud se typ písma nezdědil od stylu *Normální*, nastavte také typ *Bookman Old Style*,
 - totéž proveďte pro styly *Nadpis 2* a *Nadpis 3* (číslování dané úrovně, typ písma),
 - zajistěte, aby tento typ písma byl zvolen u stylu *Text poznámky pod čarou*.
-

1.5 Struktura dokumentu

1.5.1 Co vše má dokument obsahovat

U knih, ale i u jednodušších dokumentů, se postupně ustálila určitá struktura, kterou je vhodné dodržovat. Některé prvky této struktury jsou nepovinné, jiné vhodné pouze pro rozsáhlé knihy, další mají různou formu pro různé typy dokumentů. V následujícím seznamu jsou uvedeny běžné části struktury (postupně tak, jak jsou řazeny v dokumentu) s komentářem.

Patitul (nepovinný) se nachází na první liché stránce⁸ a obsahuje obvykle pouze název knihy, případně autora, dále nakladatelství a případně další informace jinde neuvedené. Text je sázen do horní třetiny stránky, nenápadným písmem (obvykle stejným jako běžný text v práci), nepoužíváme žádné výraznější formátování. Úkolem patitulu je pouze „částečně zaplnit“ prázdnou stránku.

Patitul se už dnes příliš nepoužívá, jen někdy u knih tištěných oboustranně⁹, a to pouze v případě, že je použit i protitul (viz dále).

Protitul (nepovinný) se tiskne na první sudou stránku – z druhé strany listu, kde je patitul nebo hlavní titul (když patitul nepoužijeme). Bývá jakýmsi grafickým doplňkem úvodní části knihy, můžeme zde umístit třeba ilustraci související s tématem publikace. Pokud tato stránka má obsahovat text, umístíme zde informace, které se nevešly na stránku hlavního titulu.

Hlavní titul je na liché stránce, tedy vpravo. Jsou zde nejdůležitější informace o knize – název, autor, nakladatelství, rok a místo vydání, případně podtitul (druhý název). Může být řešen graficky výrazněji než ostatní úvodní části.

Vydavatelský záznam (jen u publikací vydaných nakladatelstvím) je na sudé stránce hned za hlavním titulem a dodává ho nakladatelství. Jsou zde informace o autorských právech, případné varování se zákazem šíření obsahu knihy bez souhlasu, nakladatelství, místo a rok vydání, ISBN¹⁰. Pokud je publikace vydána pod svobodnou licenci¹¹, bývá zde informace o možnostech šíření dokumentu a licenci samotné.

Věnování, poděkování je na liché stránce, je jednoduše formátované.

Předmluva (nepovinná) ještě není součástí textu (nebývá ani zahrnuta do obsahu dokumentu) a nevztahuje se přímo k tomu, o čem dokument je, ale spíše o publikaci samotné (komu je určena, jak a proč vznikla, apod.), často ji píše někdo jiný než autor (překladač, vydavatel). Začíná na liché stránce.

⁸Lichá stránka je stránka vpravo, sudá je vlevo. „První“, vlastně nultá, stránka vlevo je vlastně zadní strana předních desek, ta se nečísluje.

⁹Semestrální, seminární a diplomové práce se tisknou jednostranně, tedy na každý list jen jedna strana, i když na některých školách se už povoluje oboustranný tisk.

¹⁰ISBN (International Standard Book Number) je desetimístné identifikační číslo, které se přiřazuje tisknutým knihám (tedy ne například diplomovým pracím, protože nejsou vydávány žádným nakladatelstvím). Tato čísla jsou celosvětově jednoznačná, žádné dvě knihy nemají stejné ISBN.

¹¹Dokument vydaný pod některou svobodnou licenci lze za určitých podmínek volně šířit (včetně kopírování). Příkladem je FDDL (Free Document Dissemination Licence), která dovoluje dokument volně ukládat, kopírovat, překládat, apod.

Obsah se v beletrii řadí na konec knihy (nebo vůbec není zařazen), ale v odborných publikacích na začátek před Úvod. Začíná vždy na liché stránce. Při větším množství tabulek nebo obrázků (s číslováním a popisky, viz kap. 1.6.2) zde zařadíme také *Seznam tabulek*, resp. *Seznam obrázků* (případně jiných typů objektů – algoritmů, grafů apod.).

Úvod se sází také od liché stránky, nebývá (ale může být) číslovaný a narozdíl od Předmluvy se týká přímo obsahu dokumentu. U odborných publikací zde píšeme úvodní informace o tématu dokumentu, motivaci (proč se tématem zabývat, proč je zajímavé), vysvětlení základních pojmů, případně historii a vývoj problematiky, informace o obsahu jednotlivých kapitol – kde co najdeme, atd. Pokud do dokumentu nezařadíme Předmluvu, můžeme to, co bychom jinak do ní napsali, přidat do Úvodu.

Textová část je popsána dále v kapitole 1.5.2.

Přílohy (Dodatky) jsou jakési „doplňující“ kapitoly, které se vztahují k tématu dokumentu, ale z nějakého důvodu (například pro přílišnou rozsáhlost, podrobnost, apod.) je neřadíme přímo do textové části. Jsou to například statistické údaje, které zabírají více místa, seznamy, velké tabulky či obrázky, doplňující informace pro „dobrovolné“ rozšíření obzorů čtenáře, apod.

Přílohy jsou obdobou kapitol textové části, ale narozdíl od nich se číslují velkými písmeny. První příloha je místo 1 označena A, druhá B, atd. Příloha může být také členěna ve více úrovních, další úrovně jsou však číslovány čísly (například podkapitola B.1.8 je v druhé příloze).

Závěr (Doslov) je shrnutím obsahu a přínosu knihy, autor zde hodnotí přínos své práce. Závěr je jakýmsi zrcadlem Úvodu, je podobně formátován, a pokud kapitola Úvod není číslována, totéž platí i o kapitole Závěr. Stejně jako Úvod, i Závěr se řadí do obsahu a sázíme ho na liché stránce (tj. vpravo).

Literatura (Bibliografie) je v odborných dokumentech včetně diplomových prací povinná, obsahuje seznam použité a doporučené literatury. Podrobněji se na její vytvoření podíváme v kapitole 1.7 o dokumentových seznamech.

Rejstřík (nepovinný) je seznam důležitých (klíčových) slov s uvedením stránky, na které je dané slovo uvedeno nebo vysvětleno. Rejstříku se také budeme věnovat v kapitole 1.7.

Tiráž (jen u publikací vydaných nakladatelstvím) obsahuje technické údaje o knize – kromě názvu, autora, nakladatelství a ISBN také například tiskárnu, typ písma, rozsah knihy apod. Stejně jako Vydavatelský záznam ji přidává nakladatelství.

Nyní si upřesníme, které části bývají v různých typech dokumentů použity:

1. *Diplomová a ročníková práce* se u většiny vysokých škol tiskne jednostranně (tedy na listu je potištěna jen jedna stránka, záhlaví a zápatí jsou jednotná), obvyklý rozsah je kolem 30 stran u bakalářské a ročníkové a kolem 50 stran

u magisterské práce. Požadavky na tento typ prací na některých vysokých školách jsou v seznamu literatury na odkazech označených písmenem „D“.

Dokument obsahuje tyto části:

- hlavní titul – název školy, fakulty a katedry, autor, typ práce, název, vedoucí práce, místo a rok,
- některé školy zde požadují abstrakt práce (popis tématu práce v rozsahu asi poloviny strany), a to v češtině (abstrakt) i v angličtině (abstract),
- prohlášení o tom, že student práci vypracoval samostatně a použil pouze uvedené zdroje, formátuje se stejně jako následující poděkování,
- poděkování (vedoucímu práce a případně dalším osobám),
- obsah (včetně položek pro přílohy!), případně seznam obrázků či tabulek,
- úvod, číslované kapitoly textové části práce, kapitoly první úrovně začínají vždy na nové stránce, závěr (může být až za přílohami),
- přílohy (pokud jsou),
- seznam použité literatury.

2. *Seminární a semestrální práce* jako praktický výstup některého předmětu jsou jednodušší, obvykle obsahují:

- hlavní titul – autor, název, typ práce a předmět, do kterého je práce dělána, může být název školy, fakulty a katedry, místo a rok; u méně rozsáhlých prací (tak do 10 stránek) není třeba pro hlavní titul vyhradit celou stránku,
- obsah (pokud máme více než jen pár kapitol a hlavní titul je na samostatné stránce),
- číslované kapitoly textové části, u méně rozsáhlých prací hlavní kapitoly nemusí začínat na nové stránce, totéž platí pro přílohy (pokud jsou),
- seznam použité literatury (pokud kapitoly první úrovně začínají vždy na nové stránce, totéž musí platit i pro seznam použité literatury).

3. *odborné články* mívají rozsah asi tak 5-20 stránek. Většinou je pro ně formát předepsán a autor má k dispozici také ukázkový dokument s popisem formátu. Články mívají tuto strukturu:

- hlavní titul – název práce, autor (-ři), pracoviště nebo v případě studentů škola, kontakty (e-mail),
- abstrakt (odstavec stručně shrnující cíle práce) sázený většinou ve zúženém odstavci menším písmem než zbytek textu,
- číslované kapitoly textové části práce následující přímo za sebou (tj. před hlavní kapitoly nikdy nevkládáme konec stránky) včetně úvodní a závěrečné kapitoly,
- seznam použité literatury.

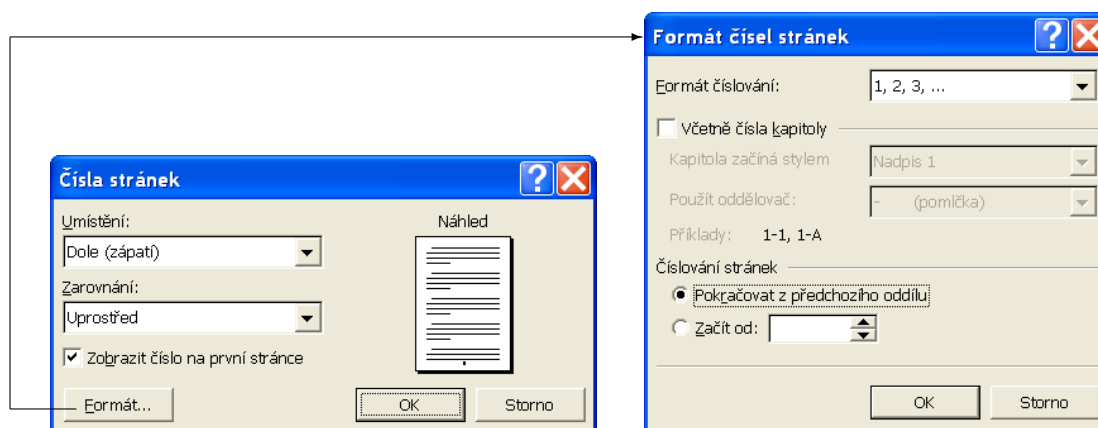
Stránky dokumentu musí být také číslovány. Čísla stránek jsou na všech stránkách *kromě* stránek prázdných, hlavního titulu, patitulu, protitulu, vydavatelského záznamu, věnování a tiráže.

Způsobem číslování můžeme odlišit oblast před textovou částí (tedy především to, co se neobjeví v obsahu), tedy stránky s předmluvou a obsahem mohou být číslovány třeba římskými číslicemi (I, II, III, IV, ...), v tom případě se změnou způsobu číslování začínáme zase od čísla 1 (jak je to i v tomto dokumentu – Předmluva a Obsah jsou číslovány římskými číslicemi a textová část arabskými od 1).

Čísla stránek se v rozsáhlejších textech umísťují většinou nahoře u vnějšího okraje (na liché stránce vpravo, na sudé vlevo). Na stránce, kde začíná hlavní kapitola (pokud je výrazněji graficky formátovaná), může být číslo umístěno jinde, přesněji celé záhlaví zůstává prázdné (například zde je číslo stránky dole uprostřed). Výchozí nastavení u většiny typografických systémů je na stránce dole uprostřed, to je však vhodné spíše pro beletrii nebo dokumenty menšího rozsahu.

V programu MS Word máme dvě možnosti, jak zajistit vložení čísel stránek:

- zobrazíme okno Záhlaví a zápatí (viz kapitola 1.5.3 na str. 21) a použijeme zde rozbalovací seznam nebo tlačítko,
- v dialogu Čísla stránek podle obrázku 1.7 (Vložit → Čísla stránek), přes tlačítko Formát nastavíme formát číslování.



Obrázek 1.7: Vložení čísel stránek

1.5.2 Textová část dokumentu

Textová část je u odborných publikací členěna do číslovaných kapitol ve více úrovních (obvykle třech – kapitoly, sekce a podsekce, druhou a třetí úroveň můžeme souhrnně nazývat podkapitoly).

Text členíme do kapitol a podkapitol tak, aby nerozdělené části textu byly tematicky souvislé a zhruba stejně rozsáhlé (jedna až několik stran, jak se „podaří“), samozřejmě s výjimkami těch podkapitol, které se tematicky rozdělit nedají.

U dokumentů v rozsahu větším než přibližně deset stran (hranice je neostrá) hlavní kapitoly začínají vždy na nové stránce, u hodně rozsáhlých dokumentů nebo u beletrie navíc hlavní kapitola začíná vždy na liché stránce.

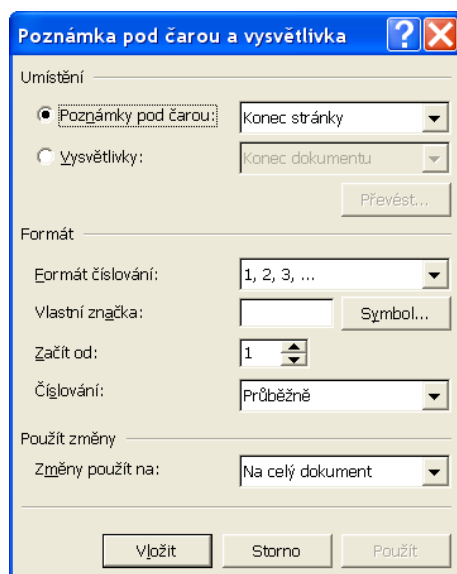
U rozsáhlých dokumentů může být dokument členěn na *části*. Části dokumentu pak tvoří hlavní strukturu, každá je věnována vlastnímu tématu (jakási „kniha

v knize“), kapitoly mohou (nemusí) být číslovány průběžně (u průběžného číslování neplatí, že by se začátkem nové části muselo být číslování kapitol nastaveno zase na 1). Každá část knihy má obdobu hlavního titulu, která se nazývá *mezititul*. Ten obsahuje jednoduše název části, případně zde mohou být uvedeny informace týkající se pouze této části včetně „částečného“ obsahu.

Ukázkou dokumentu s rozdělením na části je například kniha *Linux dokumentační projekt*, která je ke stažení na stránkách nakladatelství Computer Press ve formátu PDF ([odkaz](#) – má více než 16 MB!), je šířená pod licencí FDDL.

Kratší texty mohou být sázeny do *sloupců*. Používání sloupců je typické pro noviny, časopisy nebo sborníky (knihy publikující odborné články). Při používání sloupců musíme řešit různé problémy související s omezením místa, například umístění obrázků a tabulek. Takové objekty se proto obvykle zabírají šířku několika sloupců. V programu MS Word se sloupce nastavují v menu Formát → Sloupce, kde určíme počet a šířku jednotlivých sloupců.

V odborných textech používáme také *poznámky pod čarou a vysvětlivky*. Obojí slouží k rozvedení nebo ujasnění pojmu uvedeného v textu. Poznámky pod čarou se sázejí na spodní okraj téže strany jako pojem, ke kterému patří (může pokračovat na spodním okraji následující strany) a od textu se oddělují několikacentimetrovou čarou, vysvětlivky se řadí na konec kapitoly nebo textové části publikace. V MS Wordu vkládáme obojí přes Vložit → Odkaz → Poznámka pod čarou, viz obrázek 1.8 na straně 21.



Obrázek 1.8: Vložení poznámky pod čarou nebo vysvětlivky

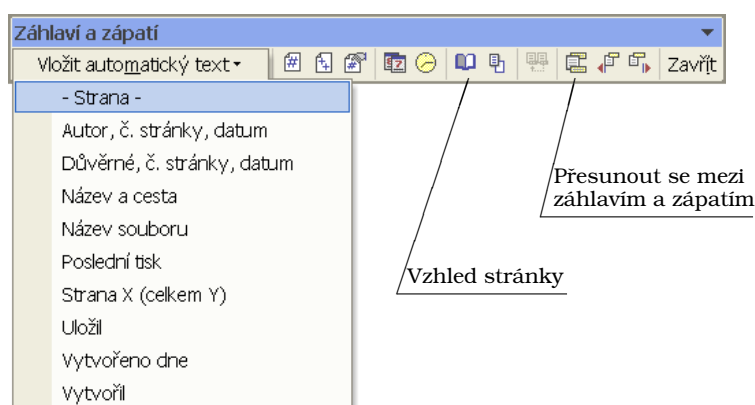
1.5.3 Záhloví a zápatí

Záhloví (hlavička) je oblast nad textovou částí stránky, zápatí (pata) pod textovou částí. Rozlišujeme záhlaví (a také zápatí) „živé“ a „mrtvé“ – živé záhlaví obsahuje

informace, které se během listování mění (například název kapitoly a podkapitoly, číslo stránky), mrtvé se nemění (název knihy, autor, obrázek-logo).

Do záhlaví odbornějších publikací jednostranných (kde se nerozlišuje pravá a levá stránka) umístíme vlevo název hlavní kapitoly a vpravo číslo stránky, u publikací oboustranných je na vnějších okrajích číslo stránky a na vnitřních název hlavní kapitoly (vpravo, na liché stránce) a název podkapitoly druhé úrovně (vlevo). Do zápatí většinou moc informací nedáváme, abychom opticky nenarušili spodní okraj textu, zvláště pokud vkládáme poznámky pod čarou. Pouze do zápatí stránky, na které začíná nová kapitola, můžeme dát číslo stránky.

Se záhlavím a zápatím v MS Wordu pracujeme pomocí dialogu Záhlaví a zápatí, který získáme v menu Zobrazit → Záhlaví a zápatí. Pokud už záhlaví něco obsahuje dostaneme se k němu také poklepáním na jeho obsah, totéž platí o zápatí.



Obrázek 1.9: Nastavení záhlaví a zápatí

V dialogu získaném přes tlačítko Vzhled stránky nastavujeme také to, zda má být dokument jednostranný nebo oboustranný (volba Různé liché a sudé) a jestli má být záhlaví a zápatí jiné na první stránce dokumentu nebo oddílu (viz kapitola 1.5.4, v samostatných oddílech můžeme mít třeba různé kapitoly). Přímou tvar a obsah již určujeme přímo v konkrétním záhlaví.

1.5.4 Oddíly

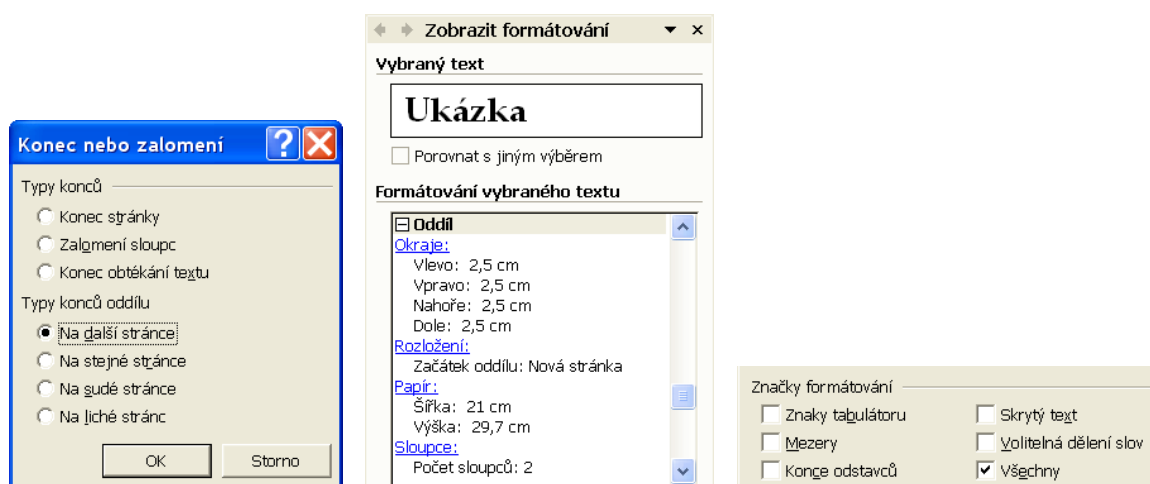
Oddíl je část dokumentu s vlastním určením formátování. Oddíl vytváříme, pokud chceme mít v takové části třeba jiné číslování, počet sloupců, záhlaví a zápatí, okraje, orientaci papíru, číslování poznámek pod čarou apod. Typicky použijeme první oddíl bez číslování stránek pro první strany dokumentu a další oddíl pro strany číslované, můžeme také použít nový oddíl pro každou další hlavní kapitolu (budou mít různá záhlaví).

V programu MS Word je v nově vytvořeném dokumentu zatím jediný oddíl. Pokud chceme přidat další, vložíme konec předchozího oddílu a tak automaticky vytvoříme oddíl následující. Konec oddílu vkládáme v dialogovém okně Konec nebo zalomení z menu Vložit → Konec (viz obr. 1.10 na str. 23), tam jen ve spodní části okna určíme, kde má být hranice mezi předchozím a následujícím oddílem (na další stránce, hned za obsahem předchozího, na liché nebo sudé stránce).

Pak už stačí nastavit vlastnosti nově vloženého (nebo už dávno existujícího) oddílu. To provedeme jednoduše tak, že se v dokumentu přímo do oddílu přesuneme a formátujeme. V některých případech musíme programu „sdělit“, že nastavované vlastnosti platí pouze pro tento oddíl a ne pro celý dokument, jak je to například u číslování stránek (je popsáno v kapitole 1.5.1 na str. 19 a v kapitole 1.5.3 na str. 22).

Pokud chceme vidět, kde jsou hranice jednotlivých oddílů, zapneme zobrazování všech značek formátování. To se dělá tak, že v menu Nástroje → Možnosti na kartě Zobrazení v prostřední části (Značky a formátování) zaškrtneme políčko Všechny.

V MS Wordu of verze XP se s formátováním oddílu dá pracovat také tam, kde se zobrazují informace o vlastnostech oddílu – v dialogovém okně Zobrazit formátování z menu Formát → Zobrazit formátování (viz obr. 1.10). Tam stačí klepnout na některý modrý odkaz a hned se zobrazí dialogové okno, ve kterém provedeme nastavení.



Obrázek 1.10: Práce s oddíly

1.6 Odkazy a popisky objektů

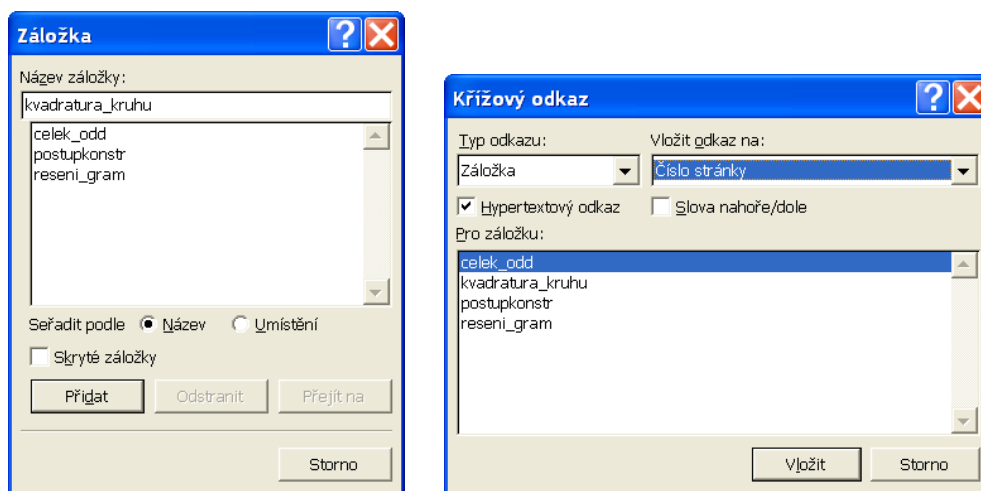
Někdy potřebujeme v textu použít odkaz na něco, co se nachází v jiné části dokumentu (například „Podrobnosti najdeme v kapitole xxx na straně yyy.“, „viz obrázek zzz.“). Samozřejmě bychom mohli údaje o kapitole a stránce napsat přímo, ale takový postup přináší jeden vážný problém – co když se při změnách obsahu nebo formátování dokumentu toto místo přesune? Uhlídat všechny podobné odkazy může být nad lidské síly. Tento problém řešíme vkládáním křížových odkazů, případně odkazů na obrázky a tabulky.

1.6.1 Křížové odkazy

Pokud potřebujeme uvést číslo stránky, na které se určitý text nachází, nejdřív musíme označit místo, na které chceme odkazovat. To se dělá tak, že slovo (slova) na tomto místě označíme (dáme do bloku) a zobrazíme dialogové okno Zložka (Vložit

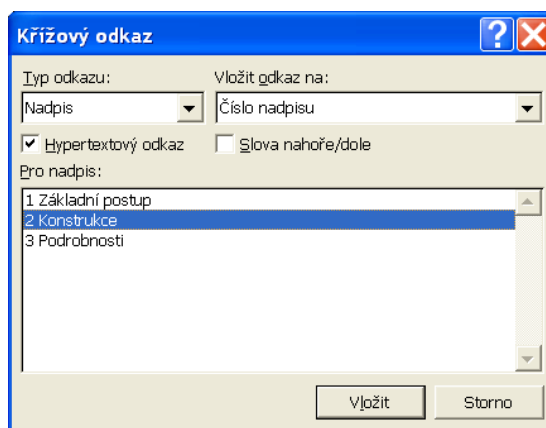
→ Zložka). Napíšeme název záložky (pozor, nesmí obsahovat mezery, a správně bychom neměli používat ani diakritická znaménka, tedy píšeme „v cestine“), viz obrázek 1.11 na str. 24. Pak klepneme na tlačítko Přidat.

Po vytvoření záložky můžeme zadat samotný odkaz. To provedeme v dialogovém okně Křížový odkaz (Vložit → Odkaz → Křížový odkaz), kde jako typ odkazu zvolíme Záložka, určíme, na co přesně chceme u objektu odkazovat (číslo stránky, na které se nachází, text odkazu apod.) a v seznamu vybereme záložku. Můžeme dokonce odkazovat na číslo položky v číslovaném seznamu.



Obrázek 1.11: Vložení křížového odkazu na místo v textu

U odkazu na číslo kapitoly nevkládáme žádnou záložku, ale v dialogu Křížový odkaz přímo určíme, že chceme odkázat na nadpis a zobrazit číslo nadpisu (v poli Typ odkazu vybereme *Nadpis*). U odkazů na nadpisy bohužel MS Word neumožňuje pracovat s jinými než předdefinovanými styly nadpisů, proto pokud chceme takové odkazy používat, nedefinujeme nové styly, ale upravujeme předdefinované.



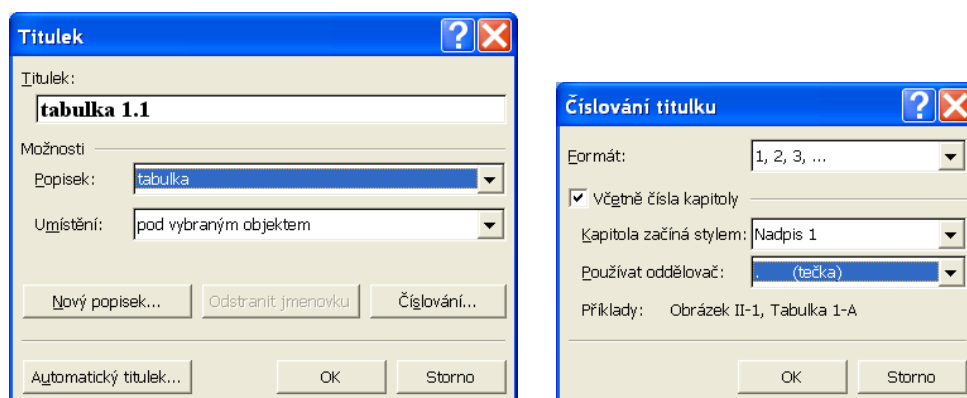
Obrázek 1.12: Vložení křížového odkazu na číslo nadpisu kapitoly

Nesmíme zapomenout, že pokud chceme odkazovat na číslo kapitoly, musí být kapitoly číslovány. Výchozí nastavení podle šablony normal.dot je takové, že nad-

pisy číslovány nejsou, takže buď změníme definici stylu, nebo u nadpisu vždy pomocí menu nastavíme číslování.

1.6.2 Obrázky a tabulky

Na obrázky a tabulky vložené do dokumentu se někdy potřebujeme odkazovat (ná-kres na obr. xxx, statistické údaje v tabulce yyy), proto takový objekt musíme označit číslem a popiskem tak, abychom odkazování mohli provádět podobně jako u klasických křížových odkazů. Popisek se umísťuje v celém dokumentu stejně, obvykle pod objekt.



Obrázek 1.13: Vložení titulků tabulky a nastavení číslování obrázků

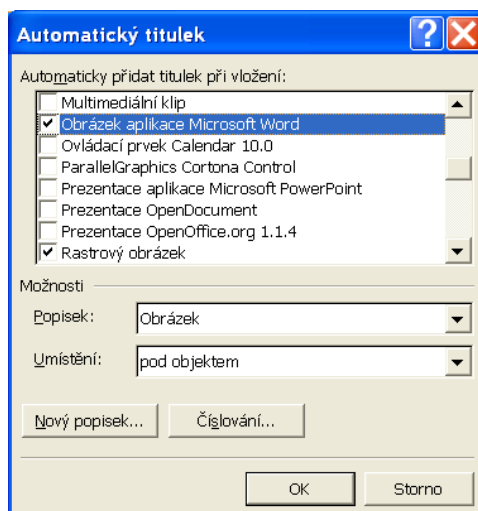
V programu MS Word postupujeme následovně:

- vložíme objekt (obrázek, tabulku, graf, apod.) a zobrazíme dialogové okno Titulek (Vložit → Titulek nebo položku Titulek v kontextovém menu objektu),
- nastavíme vlastnosti titulků (zda jde o obrázek nebo tabulku), případně pokud jsme to ještě neudělali při vložení prvního objektu, nastavíme číslování (například návaznost na číslo kapitoly – lze volit jen styly předdefinovaných nadpisů), pro dosud nepoužitý typ objektu vytvoříme nový titulek (Příklad, Ná-kres, Diagram, ...),
- potvrdíme a přímo u objektu připišeme za popisek a číslo slovní popis objektu.

Tento postup můžeme částečně automatizovat tak, že v dialogovém okně Titulek klepneme na tlačítko Automatický titulek a v následujícím dialogovém okně (viz obrázek 1.14) nastavíme automatické vkládání titulků pro určité typy objektů. Jediné, co pak budeme muset dělat ručně, bude vypisování slovního popisu objektů.

Když máme k objektu připojen titulek, můžeme na tento titulek odkazovat. To provedeme stejně jako u křížových odkazů, jen jako typ odkazu zvolíme obrázek nebo tabulku.

U odborných prací je zvykem obrázky, tabulky a další vkládané objekty do dokumentu umísťovat bez obtékání textem. Pokud však máme jen malé obrázky (menší než polovina šířky stránky), je možné nechat je obtékat textem. Měli bychom



Obrázek 1.14: Automatický titulek objektu

však dodržovat zásadu, že když obtékat, tak všechny obrázky, nebo alespoň všechny z určité skupiny (například vyznačující ikonky, viz kap. 1.2.3, strana 8. Obtékání obrázků se v programu MS Word nastavuje v kontextovém menu Formát obrázku, záložka Pozice, upřesníme přes tlačítko Upřesnit.

1.7 Dokumentové seznamy

V této kapitole budeme používat především dialogové okno Rejstřík a seznamy z menu Vložit → Odkaz → Rejstřík a seznamy.

Pod pojmem dokumentový seznam si můžeme představit jakýkoliv seznam, který se týká celého dokumentu nebo jeho podstatné části (třeba oddílu). Je to obsah, rejstřík, seznam literatury, seznam obrázků, tabulek, příkladů, rejstřík nebo třeba dílčí obsah kapitoly, který chceme umístit na první stranu této kapitoly.

Dokumentové seznamy chápeme obvykle jako kapitoly se zvláštním významem a také je tak formátujeme (zejména nadpisy a případně uvedení v záhlaví), nečíslováme je a většinu (kromě seznamu literatury a rejstříku) nezahrnujeme do obsahu.

Tyto seznamy se týkají celého dokumentu a bývají často umístěny ještě před objekty, na které mají odkazovat, a navíc tyto objekty se mohou dynamicky měnit (přidávání, mazání, změna místa apod.), proto se často setkáváme s problémem jejich *neaktuálnosti*.

Různé typografické systémy to řeší různě, v případě MS Wordu musíme pro synchronizaci (zaktualizování) seznamu zobrazit kontextové menu tohoto seznamu, zvolit položku Aktualizovat pole a určit, zda se mají aktualizovat pouze čísla stránek (u přesunů objektů) nebo celá tabulka (pokud jsme také přidávali a mazali).

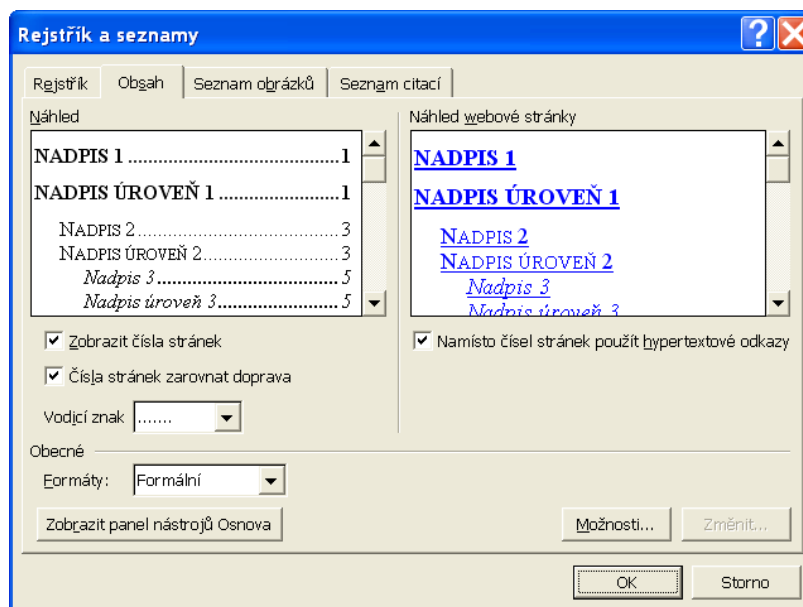
Problém neaktuálnosti se bohužel týká i různých odkazů včetně křížových. Ruční provádění aktualizace by proto znamenalo procházení celého dokumentu a aktualizování všech seznamů a jednotlivých odkazů, ale naštěstí existuje rychlejší postup – celý dokument dáme do bloku (klávesovou zkratkou **Ctrl** + **A** nebo

v menu Úpravy → Vybrat vše), zobrazíme kontextové menu bloku a vybereme položku Aktualizovat pole (objeví se pouze tehdy, když v dokumentu je co aktualizovat).

1.7.1 Seznamy podle stylů

Pod pojmem seznamy podle stylů budeme chápat cokoli, co se vytváří podle podle položek uvedených v textu formátovaných stanoveným stylem – obsah, seznam obrázků, tabulek, apod.

Obsah do dokumentu vložíme pomocí dialogového okna Rejstřík a seznamy, kde zvolíme záložku Obsah. Zde také určíme formát, počet zobrazených úrovní, v dialogu Možnosti obsahu (tlačítko Možnosti) zvolíme, které nadpisové styly se mají do obsahu zahrnout (stačí napsat číslo úrovně, do které se má styl v obsahu zařadit), a pokud máme nastavený formát Podle šablony, můžeme přes tlačítko Změnit podrobněji nastavit vzhled položek obsahu. Položky seznamů jsou také formátovány pomocí stylů, tedy jejich úpravu můžeme provádět stejně jako třeba úpravu stylů nadpisů.



Obrázek 1.15: Vložení obsahu do dokumentu

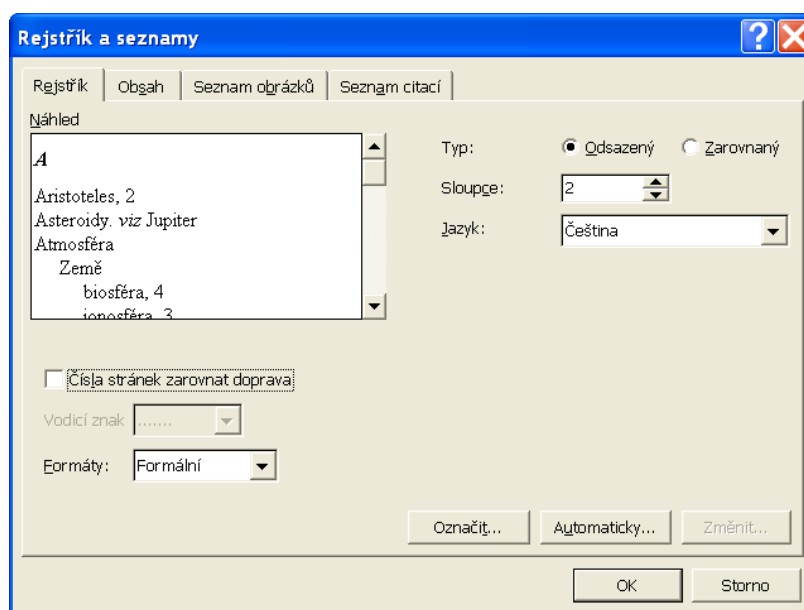
Při vkládání *seznamu podle titulků* (seznamy obrázků, tabulek, příkladů nebo čehokoliv jiného, pro co máme vytvořený typ titulků) postupujeme podobně. V dialogovém okně Rejstřík a seznamy na záložce Seznam obrázků určíme, jaký typ titulků se má do seznamu zahrnout – stiskneme tlačítko Možnosti, zvolíme styl Titulek, a pak přímo v okně Rejstřík a seznamy zvolíme jako typ titulků třeba Obrázek nebo Tabulka. Stejně jako u obsahu, i zde můžeme nastavit formát, doporučuje se použít stejný formát jako u obsahu.

1.7.2 Rejstřík

Rejstřík je seznam pojmů s odkazy na stránky, na kterých se dané pojmy nacházejí nebo jsou definovány. Používá se u rozsáhlých odborných knih, kde se předpokládá jeho využití – je zde uvedeno mnoho odborných pojmů a text je rozsáhlý, tedy rejstřík je velmi užitečný při dohledání významu pojmů. Samotný rejstřík vložíme do dokumentu přes okno Rejstřík a seznamy, záložka Rejstřík.

Na vytváření rejstříku bychom měli myslet už při psaní textu. Pojem v textu (i více slov), které chceme zařadit do rejstříku, označíme jako místo, na které má položka rejstříku odkazovat. V MS Wordu to provedeme tak, že pojem dáme do bloku a vyvoláme okno Označit položku rejstříku. Toto okno můžeme získat několika způsoby:

- použijeme klávesovou zkratku **Alt** + **Shift** + **X**,
- zobrazíme okno Rejstřík a seznamy a tam na záložce Rejstřík použijeme tlačítko Označit,
- v okně Pole (Vložit → Pole) vybereme Rejstřík a seznamy, jako název vybereme XE a klepneme na tlačítko Označit položku rejstříku,
- po úpravě zobrazených tlačítek popsané v kapitole 1.1 prostě vždy použijeme tlačítko, které jsme si přidali.



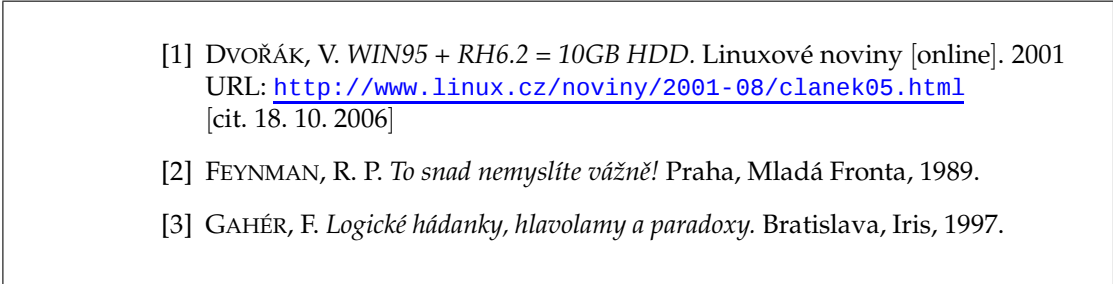
Obrázek 1.16: Vložení rejstříku do dokumentu

Při označování položek rejstříku se v některých verzích Wordu běžně stává, že se najednou začnou zobrazovat tzv. skryté znaky a jiné položky, které uživatel rozhodně neočekává. Jejich zobrazování můžeme vypnout buď stisknutím tlačítka ¶ nebo v menu Nástroje → Možnosti, kde na záložce Zobrazení v části Značky formátování odznačíme políčko Všechny.

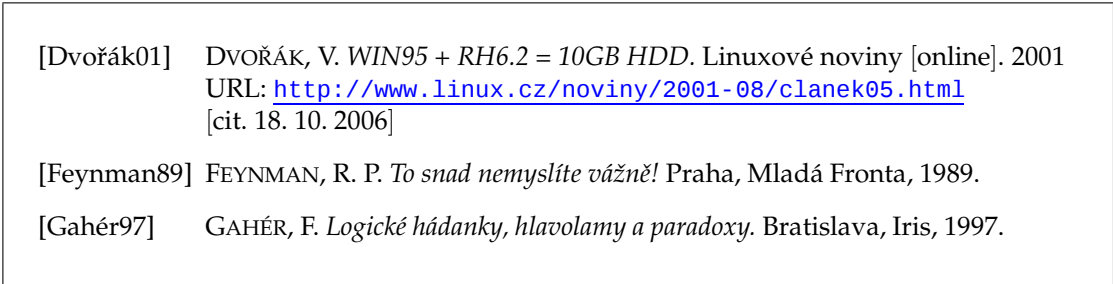
1.7.3 Seznam literatury

Seznam literatury (nebo obecně informačních zdrojů) je velice důležitá část odborné publikace. Měly by zde být zahrnuty všechny zdroje, ze kterých autor čerpal znalosti při psaní práce, a platí zásada, že by na všechny zde uvedené položky měl být někde v textu alespoň jeden odkaz neboli *citace* (alespoň v úvodní kapitole, kde odkazujeme na zdroje podrobnějších informací k jednotlivým tématům).

Před každou položkou obsahu je symbol, kterým se v textu odkazujeme na tuto položku. Může to být číslo nebo identifikační řetězec (například příjmení, zkratka příjmení, případně přidáme rok vydání). Na obrázcích 1.17 a 1.18 je ukázka dvou možností označování položek.

- 
- [1] DVOŘÁK, V. WIN95 + RH6.2 = 10GB HDD. Linuxové noviny [online]. 2001
URL: <http://www.linux.cz/noviny/2001-08/clanek05.html>
[cit. 18. 10. 2006]
 - [2] FEYNMAN, R. P. *To snad nemyslíte vážně!* Praha, Mladá Fronta, 1989.
 - [3] GAHÉR, F. *Logické hádanky, hlavolamy a paradoxy.* Bratislava, Iris, 1997.

Obrázek 1.17: Citace s odkazy ve formě čísel

- 
- [Dvořák01] DVOŘÁK, V. WIN95 + RH6.2 = 10GB HDD. Linuxové noviny [online]. 2001
URL: <http://www.linux.cz/noviny/2001-08/clanek05.html>
[cit. 18. 10. 2006]
 - [Feynman89] FEYNMAN, R. P. *To snad nemyslíte vážně!* Praha, Mladá Fronta, 1989.
 - [Gahér97] GAHÉR, F. *Logické hádanky, hlavolamy a paradoxy.* Bratislava, Iris, 1997.

Obrázek 1.18: Citace s odkazy ve formě s příjmením autora

Na danou položku se odkazujeme v textu následovně:

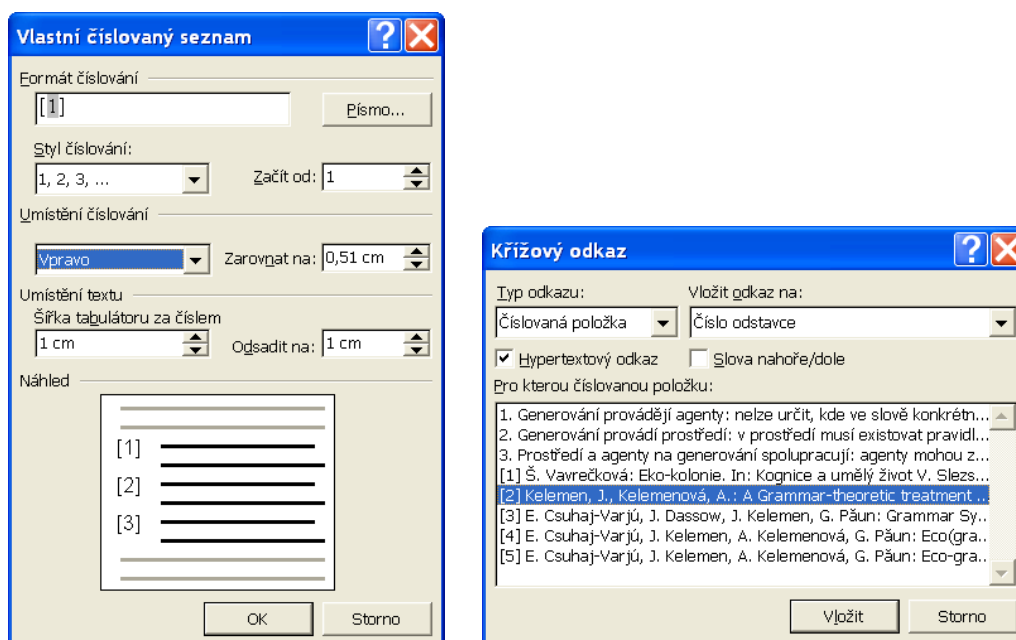
- odkaz na publikaci: „Podrobnosti najdeme v [8].“
- odkaz na více publikací: „Daná metoda je použita v [12,19].“
- odkaz na strany publikace: „Můžeme se řídit také podle [7, str. 70].“
- odkaz na rozsah stran publikace: „Postup je popsán na [14, str. 66–71].“
- odkaz na kapitolu publikace: „Doporučujeme zejména [9, kap. 5.4].“

Je možné také vytvořit seznam literatury bez označování položek (tedy bez prvního „sloupce“), pak se na tuto položku odkazujeme většinou příjmením autora a rokem v závorkách kulatých nebo hranatých – „Podrobnosti v (Nováček, 2003).“ resp. „... [Nováček, 2003]“. Výhodou oproti číslování je větší názornost pro čtenáře, nevýhodou jsou problémy v případě, že autor není znám. Tento postup se u diplomových prací nepoužívá.

Položku seznamu sestavujeme tak, aby bylo dílo dohledatelné. U běžných publikací se obvykle píše autor (autoři), název díla, místo vydání, vydavatel, rok vydání, případně ISBN a další informace. Některé typy dokumentů mají svá specifika (jako je třeba článek v časopise, odkaz jen na část dokumentu, dostupnost také internetu, dostupnost pouze na internetu, ...). Tato specifika by měla být také zohledněna, podrobnosti nalezneme v [C1, C2].

Některé části položky by měly být určitým způsobem zvýrazněny, aby byly snadno opticky vyhledatelné, především příjmení se většinou formátuje KAPITÁLKAMI a název dokumentu *kurzívou*¹², jak je to i na ukázkách na obrázcích 1.17 a 1.18. Jednotlivé části položky by měly být odděleny tečkou, čárkou, pomlčkou, dvojtečkou a středníkem, podrobnosti opět v [C1, C2].

V programu MS Word tvoříme samotný seznam jako číslovaný seznam s formátem [číslo], vhodně nastavíme také odsazení (včetně vertikálního odsazení odstavců položek). Ukázka nastavení číslování je na obr. 1.19.



Obrázek 1.19: Číslování seznamu literatury a vytvoření odkazu na položku seznamu

Odkaz na položku v textu vložíme stejně jako křížový odkaz, tedy v okně Křížový odkaz (Vložit → Odkaz → Křížový odkaz) zvolíme jako Typ odkazu číslovanou položku, v poli Vložit odkaz na vybereme číslo odstavce a pak jen v zobrazeném seznamu určíme, na co chceme odkazovat.

Odkazy na zdroje nemusíme dávat jen na konec knihy, některé (zejména internetové stránky) lze dát do poznámky pod čarou, například takto:¹³.

¹²V případě odkazu na článek v časopise nebo sborníku se takto formátuje název časopisu, resp. sborníku, tedy celkově název díla.

¹³Převzato z <http://fpf.slu.cz/~vav10ui> [cit. 28. 10. 2006].

Úlohy: _____

1. Nastavte existující styly (první tři) a vytvořte další následovně:

- *Normální* (pro písmo běžného odstavce) – typ písma *Palatino Linotype*, velikost písma 12 bodů, mezeru za odstavcem 6 bodů,
- *Nadpis 1* – písmo *Palatino Linotype*, tučné, 16 bodů, víceúrovňové číslování, pro tok textu odstavce nastavte také *Svázat řádky*,
- *Nadpis 2* – písmo *Palatino Linotype*, tučné, 14 bodů, víceúrovňové číslování, mezeru za odstavcem nastavte na 6 bodů,
- *Nadpis přílohy* – vychází z *Nadpis 1*, styl následujícího odstavce *Normální*, v číslování nastavte velká písmena bez tečky,
- *Nadpis obsahu* – vychází z *Nadpis 1*, styl následujícího odstavce *Normální*, bez číslování, v nastavení Odstavec nastavte Úroveň osnovy na *Základní text*, aby se tento nadpis neobjevil v obsahu,
- *Nadpis literatury* – totéž jako předchozí, jen necháme v odstavci Úroveň osnovy na *Úroveň 1*.

2. Titulní strana: celá bude zarovnaná na střed a takto naplněná:

- na první straně nahoře umístíte na několika řádcích název své školy a třídu (písmo 10 bodů),
- přibližně uprostřed stránky, mírně výše, bude nadpis práce „Struktura středně rozsáhlé práce odborného charakteru“ (písmo 18 bodů, tučné), na další řádek podtitul „Ukázkový dokument“ (16 bodů), na další řádek jméno a příjmení (16 bodů, odsazení od předchozího odstavce 18 bodů).
- Na spodním okraji stránky napište „Opava, 2006“.

3. Vložte další oddíl, typ oddílu na další stránce. Nastavte zarovnání bude do bloku. Vždy dávejte pozor, abyste změny nastavovali na tento oddíl, ne na celý dokument. Napište zde text (bez rámečku):

Prohlášení

Prohlašuji, že jsem svou práci vytvořil (-a) samostatně a použil (-a) jsem pouze zdroje uvedené v seznamu literatury.

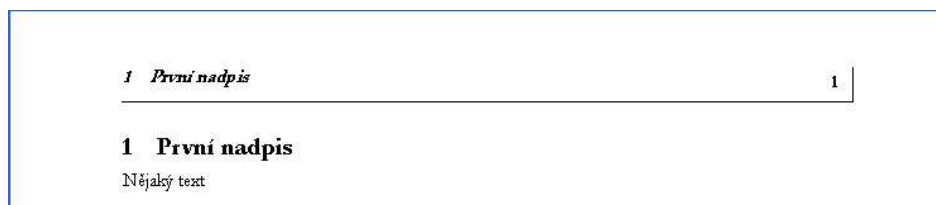
V Opavě dne 25. dubna 2007

.....
Vaše jméno

4. Vložte nový oddíl, typ oddílu na další stránce. Tam vložte nadpis „Obsah“ ve stylu *Nadpis obsahu*.

5. Vložte nový oddíl, typ oddílu na další stránce. Vložte nadpis *První kapitola* ve stylu *Nadpis 1*. Záhloví nastavte následovně: bude nezávislé na předchozím, tedy odznačte tlačítko Stejně jako minulé, vpravo bude číslo stránky (použijte volbu z hlavního menu Vložit → Číslo stránky), číslování od 1, vlevo název hlavní kapitoly (vložte *křížový odkaz* na číslo a název kapitoly, mezi tyto dva údaje

vložte speciální symbol *dlouhá mezebra*), obojí tučným písmem s kurzívou, záhlaví ohraničte způsobem naznačeným na obrázku 1.20. Napište do první kapitoly pár slov, vložte podkapitolu s jakýmkoliv názvem.



Obrázek 1.20: Ukázka formátu záhlaví

6. Vložte další oddíl a v něm druhou kapitolu nazvanou *Druhá kapitola*. Záhlaví tvořte stejně jako u předchozího oddílu, jen nechte označené Stejně jako minulé, číslování stránek navažte na předchozí oddíl a vložte číslo a název této kapitoly.
7. Vraťte se na stranu pro obsah a vložte obsah dokumentu.
8. Na konci dokumentu vložte další oddíl, ve kterém nechte navázat číslování stránek. Vložte nadpis ve stylu *Nadpis přílohy* a v záhlaví opět změňte číslo a název kapitoly. Stejným způsobem vložte ještě druhou přílohu.
9. Vložte nový oddíl stejně jako u předchozích, novou kapitolu nazvěte *Literatura* a zformátujte ji stylem *Nadpis literatury*. V záhlaví nechte pouze číslo stránky.
10. Aktualizujte obsah.

Obsah	
1	PRVNÍ NADPIS1
1.1	PODNADPIS V PRVNÍ KAPITOLE1
2	DRUHÁ KAPITOLA2
PŘÍLOHA A	PRVNÍ PŘÍLOHA3
PŘÍLOHA B	DRUHÁ PŘÍLOHA4

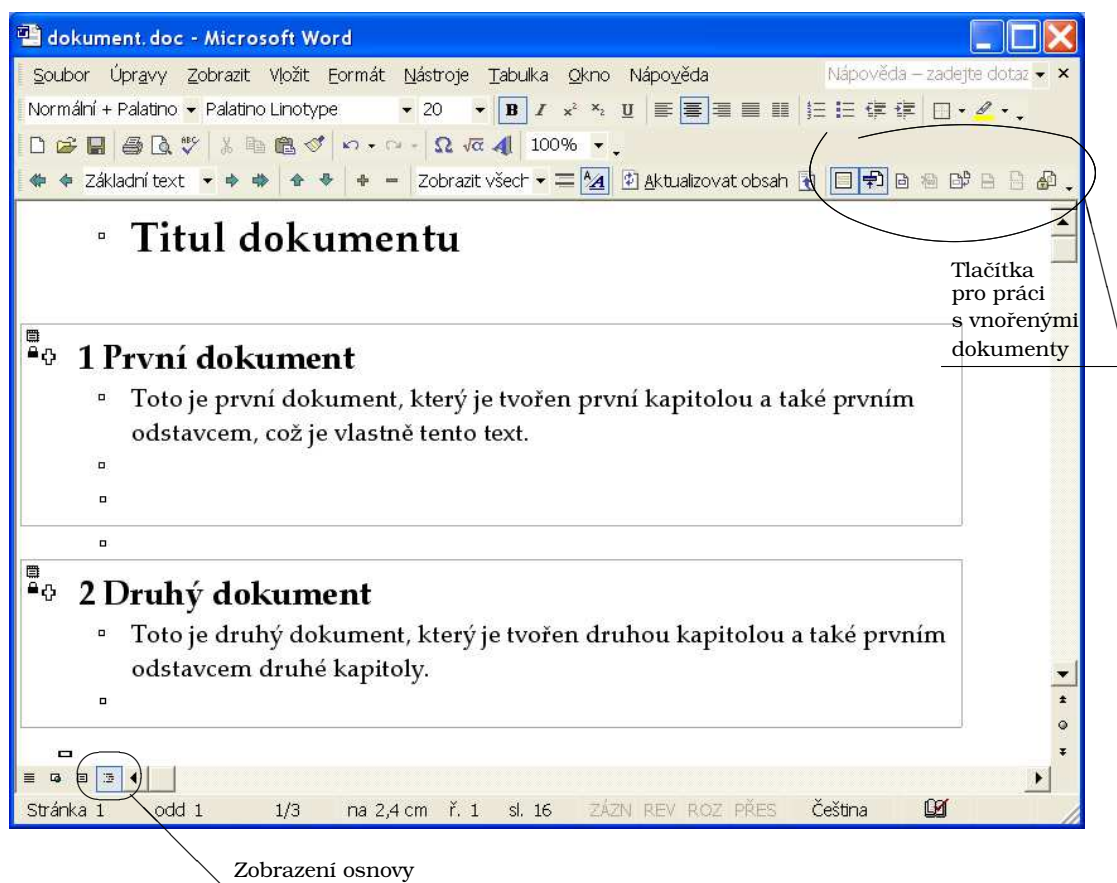
Obrázek 1.21: Ukázka formátu obsahu

11. V druhé kapitole vložte jakýkoliv obrázek a tabulku (obojí bez obtékání, můžete i více), přidejte titulky s popiskem číslovaný včetně čísla kapitoly. Za obsah (do stejného oddílu) vložte stránku se seznamem obrázků a stránku se seznamem tabulek s nadpisy *Seznam obrázků* a *Seznam tabulek* ve stylu *Nadpis obsahu*. Od obsahu můžete oddělit například vložení konce stránky.
12. Na internetu si najděte několik adres, které vás zaujaly, a odkazy na ně vložte do seznamu literatury. Připište také některou knihu, například z nakladatelství Computer Press – viz odkaz [P12]. Pro seznam zvolte vhodný číslovaný formát.
13. Do textu v první kapitole vložte odkaz na některou položku seznamu literatury.

1.8 Dlouhé dokumenty

Příliš dlouhé dokumenty mohou působit spoustu problémů. Nejde jen o problémy autora dokumentu (postupně ztrácí přehled a souvislosti), ale problémy mívá i MS Word (rozhození stránek, ztrácení objektů apod.). Řešením je rozdělit dokument do několika menších *vnořených dokumentů* a propojit je přes jeden *hlavní dokument*.

Vnořené dokumenty tvoříme stejně jako kterékoliv jiné, do každého můžeme třeba umístit jednu kapitolu. S číslováním si nemusíme dělat starosti, můžeme v každém vnořeném dokumentu nechat číslování od 1, po propojení se provede automatické přečíslování tak, aby kapitoly navazovaly. Aby se provázání podařilo, musíme dbát na dodržování stylů, včetně stylu nadpisů. Ty definujeme nejlépe v hlavním dokumentu a pak přeneseme do všech vnořených. Pokud jsme styly nastavili podle popisu v kapitole 1.4.2, bude v pořádku i umístění hlavních kapitol na novou stránku.



Obrázek 1.22: Vnořené dokumenty

Při sestavování dlouhého dokumentu postupujeme následovně:

1. Vytvoříme hlavní dokument a definujeme v něm všechny potřebné styly.
2. Vytvoříme dokument s první částí textu (třeba pro první kapitolu) a přejmeme šablonu stylů z hlavního dokumentu. Stejně vytváříme i další dokumenty.

3. Následující body můžeme provést kdykoliv – hned po vytvoření dokumentů nebo až po jejich naplnění:

- všechny dílčí dokumenty i hlavní dokument pojmenujeme a uložíme,
- v hlavním dokumentu si zvolíme zobrazení osnovy (Zobrazit → Osnova nebo jedním z tlačítek v levém spodním rohu okna Wordu),
- na panelu nástrojů (nahore na liště pod menu) se objeví tlačítka pro vkládání dokumentů, použijeme tlačítko Vložit vnořený dokument a najdeme soubor s prvním dokumentem, ze kterého se má celek skládat,
- totéž provedeme i pro ostatní dokumenty.

4. Vnořené dokumenty můžeme kdykoliv upravovat, změny se po uložení projeví i v hlavním dokumentu¹⁴.

Na liště, kde je tlačítko pro vkládání vnořených dokumentů, najdeme i další tlačítka pro práci s těmito dokumenty, hodit se může například možnost je sbalit (v hlavním dokumentu se budou zobrazovat místo celých souborů pouze odkazy na ně, dokument je pak přehlednější), před tiskem nebo náhledem bychom je měli zase rozbalit.

1.9 Formáty souborů

Každý soubor na disku má svůj vlastní formát určený programem, ve kterém byl vytvořen. Existují soubory *čistě textové* (obsahují jenom znaky – písmena, číslice, apod.), které vytváříme například v Poznámkovém bloku¹⁵, častější jsou však soubory binární.

Binární soubor obvykle musíme otevírat pouze v programech k tomu určených, jejich vnitřní struktura je totiž složitější než v případě čistě textových souborů. Jsou to například obrázky (pro ty potřebujeme grafický editor, přinejmenším Malování z Windows), ale také tabulky (třeba z Excelu) nebo například dokumenty Wordu.

U textových souborů bývají kromě samotného textu uloženy informace o formátování (typ písma, který úsek je psán kurzívou, jaké je odsazení prvního řádku odstavce, atd.), o vložených netextových objektech (obrázky, tabulky, diagramy, apod.) a také o souboru samotném (autor, verze programu, ve kterém byl soubor vytvořen, atd.). Ne každý program „ví“, jak jsou které informace uloženy, problém bývá zvláště u typů dokumentů s *uzavřeným formátem*¹⁶, navíc pokud se verze od verze tento formát mění.

To, jaký má určitý soubor formát, poznáme většinou podle jeho přípony. Ve výchozím nastavení Windows se přípony souborů nezobrazují, ale zjistíme je buď ve vlastnostech souboru (v kontextovém menu názvu souboru vybereme položku Vlastnosti) nebo si zapneme zobrazování přípon. To provedeme následovně:

¹⁴Je to z toho důvodu, že vnořené dokumenty ve skutečnosti nejsou do hlavního přímo vkládány, ale je zde použit pouze odkaz a přepínáním mezi sbalenými a rozbalenými dokumenty pouze určujeme, jak se má s tímto odkazem pracovat

¹⁵ → Programy → Příslušenství → Poznámkový blok

¹⁶Firma, která formát vyvinula, ho nezveřejnila, to je třeba případ formátu DOC pro MS Word. Opak uzavřeného formátu je otevřený formát, jehož strukturu autoři zveřejnili nebo je snadno zjistitelná.

- otevřeme některého správce souborů – například poklepeme myší na kteroukoliv složku, pokud možno tu, ve které je náš soubor, nebo třeba zobrazíme kontextové menu ikony Tento počítač na Ploše a tam vybereme položku Prozkoumat,
- najdeme náš soubor,
- v menu zvoleného správce souboru zvolíme Nástroje → Možnosti složky,
- přepneme se na záložku Zobrazit a najdeme položku Skrýt příponu souborů známých typů, kterou odznačíme (tj. nesmí zůstat zatržená).

Pro nás jsou zatím důležité tyto formáty:

DOC – dokument MS Wordu,

XLS – tabulka MS Excelu,

PPT – prezentace MS PowerPointu,

DOT – šablona dokumentu MS Office (včetně Wordu),

TXT – čistý textový dokument,

RTF (Rich Text Format) – částečně přenosný dokument určený především pro text,

PDF (Portable Document Format) – přenosný dokument firmy Adobe.

První tři formáty zřejmě zná každý uživatel MS Office, proto se jim blíže věnovat nebudeme, o souborech s příponou DOT jsme mluvili v kapitole Styly a šablony (kapitola 1.4), o čistě textových dokumentech celkem není co říct, proto se budeme věnovat zbylým – RTF a PDF.

RTF je formát, jehož specifikace (struktura) je známá a bývá používán v mnoha různých programech včetně MS Wordu¹⁷. Tam do formátu RTF uložíme dokument jednoduše – v menu Soubor → Uložit jako zvolíme ve spodní části okna položku Formát RTF. Soubor tohoto formátu může obsahovat prakticky cokoli (jen po uložení do Wordu se některá nastavení specifická pro formát DOC mohou „rozházet“), ale opravdu je určen především pro texty (narozdíl od TXT i s formátováním), proto po vložení binárního objektu (třeba obrázku) bude zabírat na disku zbytečně mnoho místa.

Soubory s příponou RTF jsou přenositelnější než DOC, proto je používáme hlavně tehdy, když víme, že soubor bude otevírán i v jiných programech než MS Word nebo dokonce v jiných operačních systémech než Windows. Bohužel tento formát přejímá některé nečnosti hlavně u delších dokumentů, především pro blízký vztah k editoru MS Word.

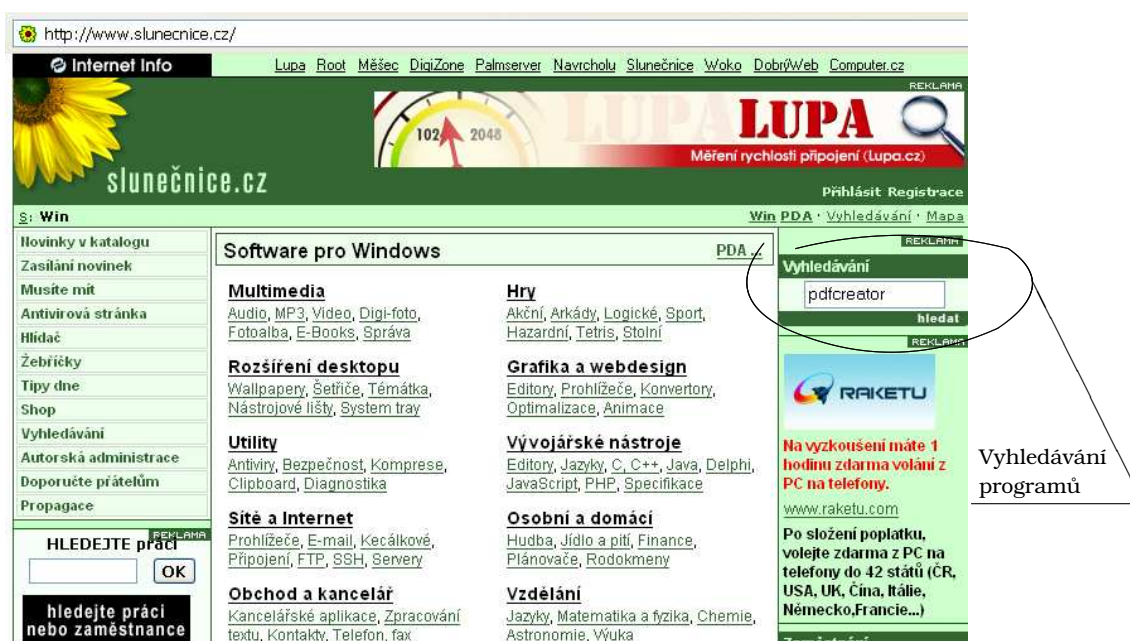
Univerzálnější a přenositelnější je formát PDF. Dokument s touto příponou (a samozřejmě vnitřním formátem) vypadá stejně, ať už ho otevřeme v kterémkoliv prohlížeči PDF souborů (Adobe Reader, PDF Reader, xpdf, apod.) nebo vytiskneme, nemůže se stát, že u dlouhého dokumentu se část nebo některé objekty „ztratí“, soubor otevřeme pod kterýmkoliv operačním systémem, kde máme některý prohlížeč nainstalován.

¹⁷Dokonce každý zručnější programátor ve Windows dokáže napsat editor, který dokáže RTF dokumenty zpracovat.

Když se nám otevření souboru nedaří, znamená to, že není nainstalován žádný prohlížeč PDF souborů. Prohlížeč autorů formátu, Adobe Reader, si můžeme stáhnout ze stránek s odkazem [P11].

PDF soubor můžeme vytvořit několika způsoby, žádný z nich však nelze použít přímo v instalaci MS Wordu. Nejlepší možnost je v tomto případě instalování jiného programu, který dokáže s MS Wordem spolupracovat. Z volně šiřitelných programů se zdá nejkvalitnější *PDFCreator*, který instalujeme následovně:

- zobrazíme stránky Slunečnice.cz (odkaz [P13]),



Obrázek 1.23: Stránky Slunečnice.cz

- zadáme do vyhledávacího okna řetězec pdfcreator, jak je na obrázku 1.23,
- v seznamu jsou kromě samotného programu také podpory pro další jazyky – ty nepotřebujeme, stačí, když stáhneme samotný program PDFCreator (klepneme na jeho odkaz),
- stáhnutý soubor spustíme a postupujeme podle instrukcí.

Po instalaci se v systému objeví nová virtuální tiskárna¹⁸ nazvaná PDFCreator, máme možnost si ji zvolit také v programu MS Word (Soubor → Tisk..., rozbalovací nabídka Název – pozor, nepoužívejte ikonku pro tisk z hlavního panelu). Když pro tisk zvolíme tuto tiskárnu, dokument se ve skutečnosti nevytiskne na papír, ale vytvoří se PDF soubor a program se zeptá, kam ho má uložit a zda dokument chceme zároveň zobrazit.

¹⁸Virtuální tiskárna je taková tiskárna, ke které nemáme „skutečný“ přístroj, je to program, který se v systému pouze chová jako tiskárna a můžeme mu v různých programech posílat data přes dialogové okno pro tisk.

Soubory s příponou PDF se vyznačují velkou přenositelností, zachováváním formátu za jakýchkoliv okolností, a navíc jsou hůře pozměnitelné než třeba DOC soubory. Proto se dokumenty na internetu zveřejňují právě v tomto formátu.

Úlohy: _____

1. Najděte na webu Slunečnice.cz další programy pracující s formátem PDF (pro vyhledávání zadejte výraz PDF). Podívejte se, jakým způsobem pracují a pod kterou licencí jsou distribuovány.
 2. Zjistěte, zda je v systému nainstalován program PDFCreator (v seznamu tiskáren nebo v kterémkoliv programu, z něhož lze tisknout – Soubor → Tisk).
 3. Vytvořte v programu MS Word jakýkoliv dokument obsahující různě formátovaný text a styly nadpisů a uložte ho postupně v těchto formátech: DOC, PDF, RTF. Porovnejte velikost těchto souborů.
 4. Proveďte totéž co v předchozím bodu, ale použijte dokument, ve kterém je několik obrázků (vložit jeden klipart a jeden obrázek ve formátu BMP, GIF, JPG nebo PNG¹⁹).
-

¹⁹Obrázky lze získat například na <http://google.com> [cit. 9. 11. 2006].

KAPITOLA 2

Typografický systém T_EX

V této kapitole se budeme zabývat typografickým systémem T_EX a především jeho nástavbou L^AT_EX, ale jen v základních rysech. Struktura obsahu kapitoly je podobná jako v kapitole o MS Wordu – typografická pravidla již vysvětlovat nebudeme, ale ukážeme si, jak v L^AT_EXu provést to, co jsme dělali v MS Wordu.

Čtenář se nemusí obávat přílišné náročnosti této problematiky. Jednotlivé postupy jsou popisovány jako jakási „kuchařka“ a ukazovány na příkladech. Je používán editor, který sice pracuje pouze s čistým textem, ale vyznačuje klíčová (důležitá) slova a téměř vše ze základů zde můžeme provádět klepáním na tlačítka panelů.

Narozdíl od předchozí kapitoly zde nenajdeme tolik úloh. Předpokládá se, že čtenář si znalosti procvičuje na uvedených příkladech.

2.1 Jak to všechno funguje

2.1.1 Princip překládaných dokumentů

MS Word a programy jemu podobné jsou označovány zkratkou WYSIWYG – *What You See Is What You Get*, tedy dostaneš to, co vidíš. Vidíme zformátované stránky s uplatněním stylů, zobrazenými tabulkami, obrázky apod., a totéž dostaneme, když dokument vytiskneme. Ono to až tak doslova někdy nebývá, ale to je zase trochu jiný problém. WYSIWYG programy mají hodně výhod z hlediska koncového uživatele (toho, kdo dokument vytváří), ale také hodně nevýhod plynoucích většinou z toho, že program musí pořád „myslet za uživatele“, někde uvnitř mít uloženy informace o tom, jak má co vypadat a pořád to tak zobrazovat, pamatovat si pozici v dokumentu a neustále „graficky“ reagovat na pokusy uživatele o změny v dokumentu.

Tohle všechno uhlídat a ještě k tomu nabízet možnost „uživatelsky přítulného“ přístupu ke všem možnostem nastavení formátů je velmi náročné a dá se to zvládnout jen buď vysokou spotřebou zdrojů počítače a složitostí prostředí programu, nebo omezením nabízených funkcí, nebo tolerancí občasných chyb.

Program $\text{T}_{\text{E}}\text{X}$ (čteme [tech]) a jeho nástavba $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ (čteme [latech] nebo [lejtech]) jdou jinou cestou. Vlastně se nejedná o žádný editor, je to pouze sada překládacích programů, které ze zdrojového souboru s příponou TEX vyrobí cílový soubor s příponou DVI, PS nebo PDF. Zdrojový soubor můžeme vyrobit v prakticky jakémkoliv textovém editoru, který dokáže ukládat čistý text (třeba v Poznámkovém bloku), ale lepší je využít některý editor přímo k tomu určený, s tlačítky pro nejčastější akce.

Ve zdrojovém souboru vyznačujeme formátování a styly pomocí *příkazů*. Existuje například příkaz, který určitý blok písmen (slov) zformátuje jako kurzívu, nebo příkaz pro vysázení nadpisu určité úrovně. Jiné příkazy slouží k „provozním účelům“ – sdělujeme překládacímu programu, že se má použít česká znaková sada, jak se mají formátovat odstavce, jaký základní styl chceme pro dokument použít a kde přesně v souboru začíná a končí vlastní dokument.

Aby se nepopletly příkazy a samotný text, každý příkaz začíná znakem \ a jeho parametry (upřesňující informace) se dávají do složených {} (u povinných parametrů) nebo hranatých závorek [] (u nepovinných parametrů).

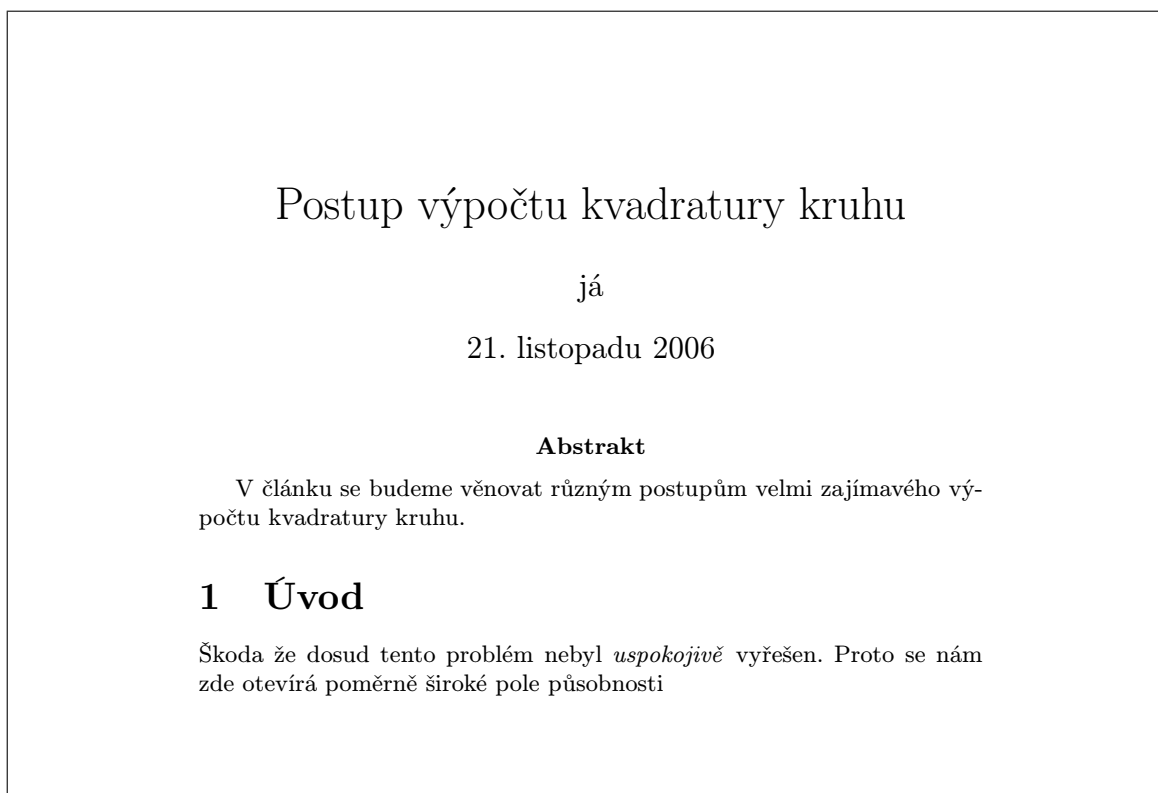
Práce v $\text{T}_{\text{E}}\text{X}$ u a $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ u obnáší zhruba takový postup, jako když sedíme u počítače s někým jiným, koukáme mu přes rameno a diktujeme: *Napiš článek v češtině, kde nadpis bude „Postup výpočtu kvadratury kruhu“, autor budu já, datum tam dej dnešní (nevím, kterého je, zaříd' to nějak), v abstraktu napiš větu „V článku se budeme věnovat různým postupům velmi zajímavého výpočtu kvadratury kruhu.“, v kapitole s názvem Úvod bude tento text: „Škoda že dosud tento problém nebyl“, teď vyznačeně „uspokojivě“, konec vyznačení, dále „vyřešen. Proto se nám zde otevírá poměrně široké pole působnosti“ atd.*

Výpis č. 2.1 ¹

<code>\typdokumentu{článek}</code>	<code>\documentclass{article}</code>
<code>\jazyk{čeština}</code>	<code>\usepackage{czech}</code>
<code>\název{Postup výpočtu kvadratury kruhu}</code>	<code>\title{Postup výpočtu kvadratury kruhu}</code>
<code>\autor{já}</code>	<code>\author{já}</code>
<code>\datum{\dnes}</code>	<code>\date{\today}</code>
<code>\začátek_dokumentu</code>	<code>\begin{document}</code>
<code>\tady_bude_titul</code>	<code>\maketitle</code>
<code>\abstrakt{V článku se budeme věnovat různým postupům velmi zajímavého výpočtu kvadratury kruhu.}</code>	<code>\abstract{V článku se budeme věnovat různým postupům velmi zajímavého výpočtu kvadratury kruhu.}</code>
<code>\kapitola{Úvod}</code>	<code>\section{Úvod}</code>
Škoda že dosud tento problém nebyl <code>\vyznač{uspokojivě}</code> vyřešen. Proto se nám zde otevírá poměrně široké pole působnosti	Škoda že dosud tento problém nebyl <code>\emph{uspokojivě}</code> vyřešen. Proto se nám zde otevírá poměrně široké pole působnosti
<code>\konec_dokumentu</code>	<code>\end{document}</code>

¹Všimněte si, že u jednoduchých dokumentů jde vlastně jen o to – znát pár anglických slovíček.

Na výpisu 2.1 můžeme vidět způsob uvažování uživatele (v prvním sloupci) a vlastní zdrojový kód dokumentu, na obrázku 2.1 výsledný vygenerovaný dokument.



Obrázek 2.1: Ukázka jednoduchého dokumentu

Jak vidíme, nemusíme vůbec myslet na typ, řez a velikost písma různých částí titulu, formát abstraktu, číslování kapitol (pak není problém kapitoly „přehazovat“), způsob formátování vyznačeného textu – vše je nastaveno podle typografických pravidel. Prakticky do čehokoliv je možné zasahovat, ale většinou to nebývá potřeba. Problém by snad mohl být se zapamatováním příkazů (to nás však nemusí trápit, pokud máme vhodný editor) a s tím, že u příkazů je potřeba dávat pozor na velká a malá písmena – když místo `\date` napíšeme `\Date`, je to chyba.

Používáme zde dva různé pojmy – T_EX a L^AT_EX. Jaký je mezi nimi rozdíl? Jde především o různé sady příkazů. L^AT_EX je nástavbou T_EXu, která má zjednodušit vytváření běžných typů dokumentů, a tak kromě původních příkazů T_EXu máme k dispozici hodně dalších, které bychom jinak museli tvořit složitě pomocí sekvencí původních příkazů. V textu dále bude používán pouze L^AT_EX.

T_EX (a L^AT_EX) není jenom jeden. Existují tzv. T_EXovské distribuce, což je vždy sada samotného T_EXu, pomocných nástrojů a především mnoha rozšiřujících *balíčků*².

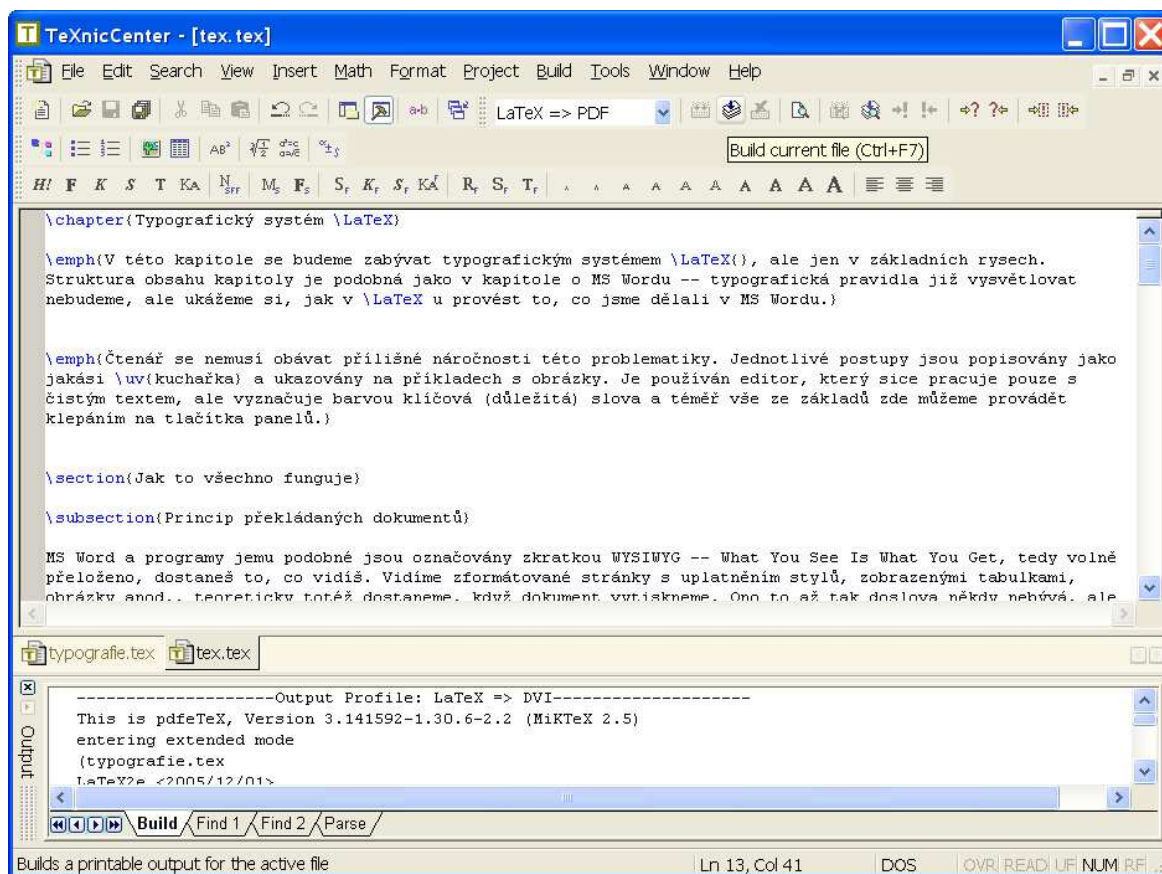
²Třeba balíček *czech* pro české prostředí, další balíčky s přednastavenými styly pro různé typy dokumentů, používání barev, fonty, vlastní definování záhlaví a zápatí stránky, speciální symboly pro různé vědní obory, atd.

U nás jsou pro Windows zřejmě nejoblíbenější distribuce MikTeX a TeXLive, na Unixových systémech je hodně používán eTeX.

2.1.2 Jak vytvořit zdrojový soubor

Zdrojový soubor, jak je výše uvedeno, můžeme tvořit v téměř jakémkoliv textovém editoru. Ideální je použít takový editor, který má v sobě (především v menu a tlačítkách) zabudovanou podporu práce v TeXu a LaTeXu.

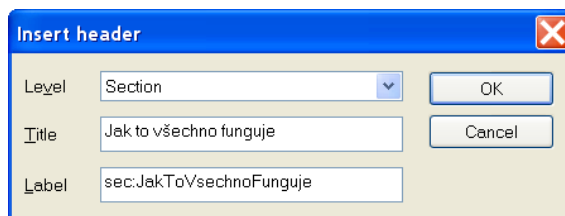
Takových programů je poměrně hodně – namátkou TeXnicCenter, TeXShell, TeXEdit, WinShell, WinEdt, ToolsCenter, EPM, Crimson Editor, PSPad, TeXMaker, MED, LyX, ... Jeden z mnoha seznamů oblíbených editorů lze najít na [T30]. Každý si může vybrat ten, který mu nejvíc vyhovuje, většinou jde o volně šiřitelné programy (až na některé – například WinEdt je shareware, tedy oproti plné verzi má některá omezení), najdeme je většinou na internetu.



Obrázek 2.2: Prostředí programu TeXnicCenter

Na obrázku 2.2 vidíme prostředí jednoho z editorů, TeXnicCenter. Na horním okraji okna máme při ruce různé nástroje, které bychom mohli potřebovat (lze měnit podle vlastních potřeb), další jsou přístupné v menu. Dále máme k dispozici dialogová okna, do kterých můžeme vyplňovat údaje potřebné k automatickému

vygenerování některého příkazu. Ukázka jednoho takového dialogového okna pro automatické vložení příkazu generujícího nadpis je na obrázku 2.3 na str. 42.



Obrázek 2.3: Dialog pro vytvoření nadpisu a „záložky“, na kterou se lze odkazovat

2.2 Začínáme

Předpokládejme, že máme nainstalovanu některou distribuci T_EXu a vhodný editor. Zde budeme používat MikT_EX verze 2.5 a editor T_EXnicCenter verze 7.01, ale to, co budeme dále probírat, je stejné téměř ve všech jiných používaných distribucích a v některých podobných editorech.

Samotná instalace není zrovna triviální (především nastavení češtiny), stručný postup instalace těchto verzí je na [P14].

Tak tedy máme spuštěn náš editor. Co teď? Vlastně nic, můžeme přímo začít psát náš text. Do nového souboru napíšeme třeba větu, kterou se obvykle testuje správné nastavení češtiny – když jsou všechna písmena správně „očárkováná“ a „oháčkovaná“, je nastavení vpořádku (uvedeno např. v [T31]). Samotná věta moc smyslu nedává:

Příliš žlutoučký kůň pěl ďábelské ódy.

Ještě než pošleme tento soubor dále na zpracování, musíme provést ještě jednu věc – přidat příkazy, které L^AT_EXu řeknou, o jaký typ dokumentu se jedná, že v souboru jsou znaky české abecedy a kde začíná a končí vlastní text dokumentu. Celý soubor bude vypadat tak jako ve výpisu 2.2.

Výpis č. 2.2

```
\documentclass{article}
\usepackage{czech}

\begin{document}
Příliš žlutoučký kůň pěl ďábelské ódy.
\end{document}
```

Část zdroje mezi příkazy `\begin{document}` a `\end{document}` se nazývá *tělo dokumentu*, část před příkazem `\begin{document}` se nazývá *hlavička dokumentu* neboli *preamble*. Do preamble se umísťují především informace o tom, jaké balíčky se mají načíst a jak má být co formátováno, tedy krátce řečeno, určujeme zde styl dokumentu.

Náš soubor je už „přeložitelný“. Uložíme ho s příponou TEX (to je přípona zdrojových souborů dokumentů v T_EXu a L^AT_EXu) a vybereme si formát a příponu pro výsledný dokument. Máme tyto možnosti:

PDF – tuto příponu už známe. PDF dokument zde můžeme vygenerovat bez jakýchkoliv dalších pomocných programů, narozdíl od MS Wordu. Soubor lze vytvořit přímo nebo přes následující formát PS³. Pro jeho prohlížení můžeme použít třeba volně šiřitelný program *Adobe Reader* (ke stáhnutí na [P11]).

PS – PostScript je formát příbuzný formátu PDF (dokonce má stejné tvůrce), jen je starší a soubory bývají většinou větší než u téhož dokumentu ve formátu PDF. Je vhodný zejména pro tisk, proto v tomto formátu máme dokumenty, které se především tisknou. Pro prohlížení existuje například program *GSView*.

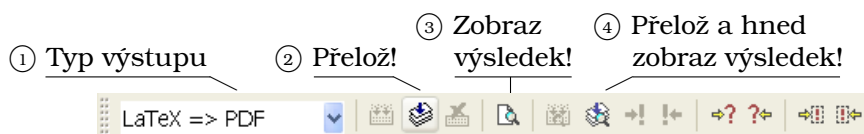
Oproti Adobe Readeru tento prohlížeč zobrazený soubor po načtení zavře a pravidelně ve velmi krátkých intervalech kontroluje, zda v souboru nedošlo ke změnám. Pro uživatele to znamená, že před dalším překladem mohou nechat dokument zobrazený (nezavírat) a navíc mají jistotu, že když se po následujícím překladu přepnou do *GSView*, vidí dokument v „čerstvém“ stavu, změny jsou aplikovány.

DVI – je to zkratka z *Device Independent*, tedy „nezávislý na zařízení“. Do tohoto formátu se soubor překládá nejrychleji, je to původní cílový formát $\text{\TeX}$ u. Obvyklým prohlížečem je *Yap* (*Yet Another Preview*, je součástí distribuce $\text{\TeX}$ u).

Formát přejímá výhody formátu PS (před dalším překladem není třeba výsledný soubor zavírat a jsou hlídány změny), navíc má výhodu rychlosti překladu. Nevýhodou je menší rozšířenost tohoto formátu, když soubor s příponou DVI zveřejníme na internetu nebo někomu pošleme, není moc velká pravděpodobnost, že adresát bude vědět, co s ním.

Kdy který formát použít? Když se chceme na výsledek podívat během práce na dokumentu, je nejlepší přeložit soubor do DVI a podívat se na dokument v *Yapu*. Je to nejrychlejší a s použitím klávesové zkratky $\text{Alt} + \text{Tab}$ máme téměř „WYSIWYG“ náhled. Hotový dokument pak zveřejňujeme v PS nebo PDF (nebo v obou formátech).

V programu $\text{\TeX}$ nicCenter používáme pro cokoliv souvisejícího s překladem panel nástrojů zobrazený na obrázku 2.4.



Obrázek 2.4: Panel nástrojů s nástroji pro překlad dokumentu

Takže naším úkolem je nyní vybrat si typ výstupu v ① (v našem případě překlad do DVI) a použít tlačítko ② nebo ④.

Prohlížeč *Yap* má oproti jiným ještě jednu důležitou výhodu – když v něm poklepeme myší na některou část dokumentu, jsme automaticky přeneseni na dané místo ve zdroji dokumentu. U tak krátkého dokumentu, jaký jsme vytvořili, to nemá moc velký význam, ale u dokumentu zabírajícího několik desítek stránek případně

³Prímé vytvoření umožňuje lépe pracovat se stránkou a jinak obcházet omezení daná formátem PS, vytvoření přes PS obvykle vytvoří menší soubor.

se zdrojem rozděleným do více souborů se to rozhodně hodí. Pokud tato vlastnost nepracuje jak má, můžeme ji nastavit v menu Yapu View → Options, záložka Inverse DVI Search (obvykle stačí jen klepnout myší na již vložený záznam, nemusíme vkládat nový).

2.3 Délkové jednotky

V kapitole 1.2.1 jsme se na straně 5 seznámili s pojmem *kuželka* a určili *stupeň písma* jako výšku kuželky v typografických bodech. Z tohoto pojmu teď odvodíme další pojmy:

čtverčik (značíme em) je míra (tj. délka) odpovídající výšce kuželky (stupni), je to přibližně výška písmene M (od toho také zkratka),

půlčtverčik (značíme ex) je polovina čtverčíku, odpovídá přibližně výšce písmene x.

Pro délky můžeme využít přímo hodnoty em a ex (například při nastavování vlastností odstavce – tyto délky se mění se změnou velikosti písma a odstavec pak vypadá „stejně“), ale používají se i další jednotky – pt (anglosaský typografický bod), in (palec), cm, mm, ... Mezi číslo a jednotku se nepíše mezera.

Například stupeň písma bývá 10pt, 11pt, 12pt, odsazení prvního řádku odstavce 2em, mezi dvěma odstavci, kde má čtenář na papíru něco doplnit, je třeba mezera 4.5cm nebo .8em (před tečkou nemusí být žádná číslice – dosadí se 0).

2.4 Písmo

Ještě než se budeme věnovat možnostem nastavení písma, musíme si ujasnit jednu důležitou věc – používání lomených závorek. Lomené závorky {} určují *skupinu*, skupiny můžeme libovolně přidávat a vnořovat. Kdekoliv máme možnost stanovit, že určitá vlastnost včetně vlastností písma bude platit od místa, kde ji určíme, až do nejbližšího konce některé skupiny.

Skupinu určují také některé příkazy, především `\begin` a `\end` (oblast ohraničená těmito dvěma příkazy se nazývá *prostředí*) – například „hlavní skupinou“ je tělo programu ohraničené příkazy `\begin{document}` a `\end{document}`.

2.4.1 Stupeň písma

Stupeň písma můžeme nastavit pro celý dokument hned na začátku, když T_EXu sdělujeme, jaký typ dokumentu chceme vyrobit:

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

Stupeň písma zde zadáváme v nepovinném parametru (je v hranatých závorkách), a to v typografických bodech (zkratka pt). Kolem závorek příkazu nepíšeme mezery, příkaz musí být „kompaktní“.



Obrázek 2.5: Panel nástrojů pro změnu stupně písma

Příkaz	Velikost
<code>\tiny</code>	velmi velmi malé písmo
<code>\scriptsize</code>	velmi malé písmo
<code>\footnotesize</code>	malé písmo
<code>\small</code>	menší písmo
<code>\normalsize</code>	normální velikost písma
<code>\large</code>	větší písmo
<code>\Large</code>	ještě větší písmo
<code>\LARGE</code>	velké písmo
<code>\huge</code>	dost velké písmo
<code>\Huge</code>	velmi velké písmo

Tabulka 2.1: Příkazy pro změnu stupně písma

Při psaní občas také chceme změnit stupeň písma. To provádíme buď na panelu podle obrázku 2.5 nebo příkazy podle tabulky 2.1 na straně 45.

Tyto příkazy nemají žádný parametr, jejich platnost je od místa, kde je napíšeme, do nejbližšího konce skupiny. Odvíjejí se od základního stupně písma, který jsme mohli určit v parametru příkazu `\documentclass` – tomu stupni odpovídá příkaz `\normalsize`.

Základní stupeň písma dále můžeme (včetně řádkování) určit přímo příkazem `\fontsize`. Informace o tomto příkazu získáme v nápovědě nebo v [T31].

Příklad 2.1 Na následující ukázce vidíme, jak na čtenáře může působit častá změna stupně písma.

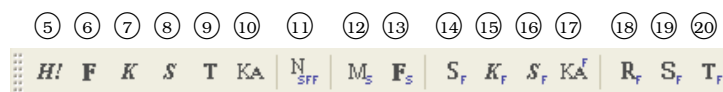
Výpis č. 2.3

`\small` Pokud chceme `\normalsize` vytvořit `{\Large` esteticky zajímavý dokument}, hodně `\footnotesize` často měníme velikosti písma. `\normalsize` Pro čtenáře to je moc dobrý způsob, jak si nacvičit `{\LARGE` ničení} dokumentu na jakýkoliv způsob.

Pokud chceme vytvořit esteticky zajímavý dokument, hodně často měníme velikosti písma. Pro čtenáře to je moc dobrý způsob, jak si nacvičit **ničení** dokumentu na jakýkoliv způsob.

2.4.2 Nastavení základních vlastností písma

Řez (*shape*), tmavost (*series*) a rodinu (*family*) písma nastavujeme jednoduše – buď použijeme tlačítka na hlavním panelu (viz obrázek 2.6) nebo přímo příkazy.



Obrázek 2.6: Panel nástrojů pro volbu řezu, tmavosti a rodiny písma

Často nám úplně stačí příkaz `\emph{vyznačený text}` pro základní vyznačení (je přednastaven na kurzívu, i když to lze změnit), kterého jsme si mohli všimnout už ve výpisu 2.1 na straně 39 – použijeme buď přímo tento příkaz nebo tlačítko  (podle obrázku 2.6 tlačítko ⑤). Příkaz lze také vnořovat, pak dostaneme text vyznačený ve vyznačeném textu, což představuje normální řez.

V tabulce 2.2 jsou vypsány různé příkazy nastavení řezů, tmavosti a rodiny písma, a to nejdřív klasickým příkazem, jehož parametrem je text pro formátování, a pak s použitím skupiny – příkaz nemusí být na začátku skupiny, ale kdekoli v ní, pak platí od místa uvedení příkazu do konce skupiny – nejbližšího znaku }.

Řez	Příkaz	Tlačítko
<i>kurzíva</i> (italic)	<code>\textit{kurzíva}</code> <code>{\itshape kurzíva}</code>	 ⑦  ⑮
<i>skloněné písmo</i> (slanted)	<code>\textsl{skloněné písmo}</code> <code>{\slshape skloněné písmo}</code>	 ⑧  ⑮
KAPITÁLKY (small caps)	<code>\textsc{kapitálky}</code> <code>{\scshape kapitálky}</code>	 ⑩  ⑰
vzpřímené (vertical)	<code>\textup{vzpřímené}</code> <code>{\upshape vzpřímené}</code>	 ⑭

Tmavost	Příkaz	Tlačítko
polotučné písmo (medium)	<code>\textmd{polotučné písmo}</code> <code>{\mdseries polotučné písmo}</code>	 ⑫
tučné písmo (bold)	<code>\textbf{tučné písmo}</code> <code>{\bfseries tučné písmo}</code>	 ⑥  ⑬

Rodina	Příkaz	Tlačítko
patkové (serif, roman)	<code>{\rmfamily patkové}</code>	 ⑱
bezpatkové (sans-serif)	<code>{\sffamily bezpatkové}</code>	 ⑲
neproporcionální (typewriter)	<code>\texttt{neproporcionální}</code> <code>{\ttfamily neproporcionální}</code>	 ⑨  ⑳

Tabulka 2.2: Příkazy pro změnu řezu, tmavosti a rodiny písma

Jak vidíme, některé příkazy jakoby neměly na písmo žádný vliv. To je proto, že například vzpřímené patkové písmo je výchozí, a dále příkaz pro polotučné písmo ve většině typů písem odpovídá normální tmavosti.

Pokud chceme zrušit všechny nastavené řezy, použijeme příkaz `\upshape` (tlačítko ⑭ – ) , když chceme zrušit nejen nastavení řezu, ale i tmavost nebo neproporcionálnost, použijeme příkaz `\normalfont` (tlačítko ⑪ – ).

L^AT_EX má také speciální způsob formátování matematických výrazů (typograficky správný). Zde se běžná velká i malá písmena abecedy sázejí skloněná nebo kurzívou a symboly, které nejsou písmeny (včetně čísel), vzpřímeným písmem, mezery mezi symboly jsou jiné než v běžném textu. Pro řízení formátu znaků zde máme k dispozici příkazy `\mathrm`, `\mathsf`, `\mathbf`, `\mathcal`, `\mathnormal`, atd.

Příklad 2.2 Ukážeme si způsob použití výše uvedených příkazů a možnosti jejich kombinace.

Výpis č. 2.4

`\textsc`{Různé} způsoby `\textbf`{vyznačení} je dobré znát, používáme jak `\textit`{tlačítka}, tak i přímo `\textit`{příkazy}.

Pro většinu vyznačení existuje možnost `\textbf`{dát vyznačovaný text do `\itshape` parametru příkazu} a druhá možnost `\bfseries` s použitím `\itshape` skupiny}.

`\emph`{Musíme dávat pozor, kde je vlastně `\emph`{konec} té skupiny, ve které chceme mít dané vyznačení!}

Příkazy `\texttt`{můžeme `\bfseries` kombinovat}, ale `\sffamily` ne vždy výsledek vypadá `\itshape` pěkně}.

`\bfseries\slshape` Pozor na mezery, `\normalfont` zvláště u příkazů využívajících skupiny!}

RŮZNÉ způsoby **vyznačení** je dobré znát, používáme jak *tlačítka*, tak i přímo *příkazy*.

Pro většinu vyznačení existuje možnost dát **vyznačovaný text do parametru příkazu** a druhá možnost **s použitím skupiny**.

Musíme dávat pozor, kde je vlastně konec té skupiny, ve které chceme mít dané vyznačení!

Příkazy můžeme **kombinovat**, ale ne vždy výsledek vypadá *pěkně*.

Pozor na mezery, zvláště u příkazů využívajících skupiny!

Za příkaz využívající hranice skupiny vždy dáváme buď mezeru nebo některé interpunkční znaménko (tečka, čárka, pomlčka, apod.), abychom tento příkaz oddělili od následujícího slova.

Před takový příkaz bychom měli dát mezeru, protože mezera (i více mezer) za příkazem se bere jako oddělovač příkazu a textu a tedy se do dokumentu nezahrnuje – tedy například počet`\itshape` žáků se vysází jako početžáků.

2.4.3 Fonty

Jak víme z předchozího textu, můžeme žádat jakékoliv změny mezi proporcionálním, patkovým a bezpatkovým písmem bez nutnosti „ruční“ změny fontu. Někdy ale potřebujeme nastavovat fonty přímo, tedy určovat typ písma dokumentu.

Narozdíl od programů typu MS Word, které využívají fonty instalované v operačním systému, nemáme v T_EXu takový výběr, používáme pouze fonty dodané přímo s T_EXem a těch rozhodně není tolik. Je to především z toho důvodu, že systém obvykle nebývá používán pro sazbu reklamních tiskovin, ale spíše odbornějších článků a knih.

Nevýhodou je tedy menší výběr, výhodou ale snadnější přenositelnost – fonty se načítají přímo do dokumentu, proto se nemůže stát, že když je dokument otevřen na jiném počítači, fonty se nezobrazí správně.

Z typografických důvodů není moc vhodné měnit fonty v různých částech dokumentu, je zvykem mít nastaven jeden a tentýž font od začátku na celý dokument. Zde si ukážeme jednoduchý způsob nastavení fontu na začátku dokumentu.

Fonty se nacházejí v balíčcích, stejně jako čeština nebo různá nastavení stylů, proto je budeme načítat stejně jako ve výpisech 2.1 a 2.2 příkazem `\usepackage`. Standardně je nastaven font *Computer Modern*. Můžeme vyzkoušet další fonty:

- palatino,
- bookman,
- helvetica,
- optima,
- times,
- garamond,
- utopia,
- avantgar (pismo AvantGarde),
- zapfchan,
- ...

Například když si vybereme první možnost, font *palatino*, napíšeme do preambule někam hned za `\usepackage{czech}` příkaz `\usepackage{palatino}`, tak jak vidíme na výpisu číslo 2.5.

Každý z těchto fontů má trochu jiné vlastnosti a možnosti použití, proto je dobré alespoň několik vyzkoušet, než si definitivně vybereme. Některé fonty nejsou definovány pro některé řezy nebo tučnost a při překladu je daná vlastnost buď pozměněna na základní, nebo nahrazena z jiného fontu.

Výpis č. 2.5

```
\documentclass{article}
\usepackage{czech}
\usepackage{palatino}

\begin{document}
Toto je ukázka dokumentu, kde používáme font
\emph{palatino}.
Tento font má zajímavější kresbu než výchozí
\emph{Computer Modern}, je výraznější.

Předtím jsme byli pořád ve stejném odstavci,
i když jsme po první větě zařadkovali,
ale teď po druhé větě jsme vložili prázdný
řádek, proto jsme vytvořili další odstavce.
\end{document}
```

Jiný způsob nastavení fontů spolu s dalšími vlastnostmi je pomocí příkazů `\usefont`, `\fontencoding`, `\fontfamily`, `\fontseries`, `\fontshape`, `\fontsize`, těmito postupy se však zde nebudeme zabývat – jsou k nalezení v nápovědě nebo třeba v [T31, str. 153].

2.4.4 Zvláštní znaky a sekvence znaků

Komentáře jsou části textu, které se ve výsledném dokumentu neobjeví. Používáme je pro vlastní okomentování textu, například informaci o tom, že na určitém místě ještě chceme něco doplnit či změnit, opticky oddělujeme různé části textu nebo potřebujeme okomentovat příkaz, o kterém si myslíme, že v budoucnu budeme mít problémy s jeho pochopením. Stačí napsat znak % (procento) a vše za ním až do konce řádku bude bráno jako komentář.

Spojovník píšeme jednoduše klávesou s tímto znakem: bude-li, česko-anglický slovník.

Pomlčka se zadává jako dva spojovníky za sebou bez mezery, tedy píšeme 10--15 minut, napsala -- pro upřesnění, zápas Sparta--Slavia. Dlouhou pomlčku pro anglické texty píšeme třemi spojovníky --- (při obklopení mezerami).

Nezlomitelná mezera taková, které je dovoleno se případně rozšiřovat, se nahrazuje vlnovkou ~, například u~domu, s~mámou.

Pevná mezera (musí zůstat malá, při posouvání slov pro rovný pravý okraj textu se nesmí rozšiřovat) se zadává příkazem \, , například 12\,450\,832, 10\,kg. Často se místo ní používá nezlomitelná mezera (~).

Vybarvený obdélník ■ se sází příkazem `\rule{šířka}{výška}` (zde jsou použity parametry `\rule{1em}{1ex}`).

Pokud zadáme některý rozměr nulový, i tak se „zabere“ místo, třebaže obdélník neuvidíme. Toho se často využívá při „nenápadném“ odsazení objektu zleva (`\rule{odsazení}{0pt}`) nebo shora (`\rule{0pt}{odsazení}`). V nepovinném parametru můžeme zadat také posun nahoru nebo dolů, takto se dá například vytvořit mezera pod textem (například `\rule[-2em]{0pt}{1pt}`).

Dlouhé horizontální mezery se zadávají příkazy `\quad` (čtverčiková) a `\qquad` (dvoučtverčiková).

Vertikální mezery jsou `\smallskip`, `\medskip` a `\bigskip` pro malou, střední a velkou mezeru mezi odstavci (například místo, kde se má něco doplnit „ručně“, s udáním délky `\vspace{délka}` – například `\vspace{2em}`).

Pružné mezery jsou `\vfill` a `\hfill`. Horizontální mezera se může použít například pro zarovnání části textu k pravému okraji (třeba číslo stránky v obsahu). Někdy je potřeba před a za tento příkaz vložit příkaz `\rule` s nulovou šířkou: `\rule{0pt}{1pt}\hfill\rule{0pt}{1pt}`.

Pružné čáry se sázejí příkazy `\hrulefill` (plná čára) a `\dotfill` (tečkovaná).

Trojtečku vyrobí příkaz `\dots`, za tímto příkazem píšeme buď některé interpunkční znaménko (třeba čárku, ale určitě ne další tečku!) a nebo, pokud má následovat mezera, musíme napsat prázdnou skupinu, aby mezera „nezmizela“, tedy píšeme `\dots{}`.

Uvozovky píšeme příkazem `\uv{text v uvozovkách}` (výsledkem je „text v uvozovkách“), úhlové jednoduše dvojicemi znaků `>>text v uvozovkách<<` (výsledkem je »text v uvozovkách«).

Procento sice najdeme na klávesnici, ale musíme použít příkaz `\%`, protože samotný znak procenta je vyhrazen pro komentáře. Píšeme 10\, \% nebo 10~\%.

T_EX, L^AT_EX, L^AT_EX 2_ε vysázíme příkazy `\TeX`, `\LaTeX`, `\LaTeXe` (to je označení zatím poslední verze systému L^AT_EX, čteme v češtině [latech dvě é]).

Stejně jako u jiných příkazů, pokud má následovat mezera, musíme za příkaz napsat prázdnou skupinu – `\TeX{}`. V případě skloňování a ohýbání slov toho využijeme – například slovo T_EXový vysázíme příkazem `\TeX` ový.

Slitky a podřezávání (viz strana 7) T_EX dělá automaticky. Pokud se nám zrovna toto chování nehodí, vložíme mezi dva znaky, které nechceme „spojit“, prázdnou skupinu (bez mezery!) – například místo `f i` bude `f{}i` a místo `fi` získáme `fi`.

Promíle, stupeň a další symboly píšeme příkazy, jejichž seznamy najdeme i na internetu, například ve zdrojích [T33] nebo [T34]. Někdy je třeba v preambuli dokumentu před příkazem `\begin{document}` uvést balíček, ve kterém jsou tyto příkazy určeny, například když je v preambuli dokumentu `\usepackage{textcomp}`, můžeme používat příkazy `\texttt{tilde}` (~), `\texttt{perthousand}` (‰), `\texttt{onehalf}` (½), nebo třeba `\texttt{musicalnote}` atd., v balíčku `gensymb` najdeme mimo jiné i příkazy `\celsius` (°C) nebo `\degree` (°).

V některých balíčcích jsou i symboly „ozdobné“ či „zábavné“, například v balíčku `marvosym` je `\Dontwash` (☒) nebo podobenka autora balíčku `\MartinVogel` (☞).

Spousta příkazů pro symboly je však dostupná bez dalších přídatných balíčků, například `\$` (\$), `\#` (#), `\{` ({}), `\&` (&), `\copyright` (©), `\texttt{trademark}` (™), `\ast` (*), `\times` (×), `\alpha` (α), atd. Mnohé najdeme v menu a na panelech editoru.

Některé zde uvedené příkazy jsou obklopeny znaky \$ (například `\ast` nebo znaky řecké abecedy). Je to z toho důvodu, že jde o symboly určené pro matematickou sazbu (v rovnicích apod.), a znaky \$ se obklopuje právě prostředí pro matematickou sazbu. To nám však nebrání tyto symboly používat i v běžném textu, stačí příkaz obklopit znaky dolaru.

Příklad 2.3 Všimněte si použití obdélníků na konci výpisu 2.6.

Výpis č. 2.6

Budeme-li v L^AT_EX u potřebovat komentář, použijeme znak `\%` -- nic za ním nebude v dokumentu vidět, a to ani ve dne (6--19~hodin). `% fakt!`

Pro násobnost používáme znak `\times`, například `3\times`, a to v matematickém prostředí ohraničeném třeba symbolem `\$`. Rovnice vložená do textu: `\$x^2-y+25=250\$`. `\hfill\rule{1ex}{1ex}`

Telefonní číslo můžeme zvýraznit symbolem `\Telefon{}` z balíčku `\texttt{marvosym}`. Tento odstavec má už 13~řádků, je třeba ho ukončit `\Rightscissors`. `\rule{.5ex}{1ex}`
`\rule{2pt}{0pt} % obdélník s nulovou výškou`
`\rule[-1ex]{.5ex}{1ex} % snížený o 1ex`

Budeme-li v L^AT_EXu potřebovat komentář, použijeme znak `%` – nic za ním nebude v dokumentu vidět, a to ani ve dne (6–19 hodin).

Pro násobnost používáme znak `×`, například `3×`, a to v matematickém prostředí ohraničeném třeba symbolem `$`. Rovnice vložená do textu: $x^2 - y + 25 = 250$. ■

Telefonní číslo můžeme zvýraznit symbolem ☎ z balíčku `marvosym`. Tento odstavec má už 13 řádků, je třeba ho ukončit ✂. ■

2.4.5 Diakritická znaménka

Diakritika obvykle nečiní problémy, lze používat i diakritická znaménka, která nemáme dostupná na klávesnici. Například `\^{}{a}` nám vysází â, `\c{o}` dává ø, příkaz `\ae` je pro æ. Podrobnější seznam je například v [T31, str. 40] nebo v [T38, str.17].

Abychom nemuseli pro nezlomitelnou mezeru tak často psát vlnku (a tedy přepínat mezi českou a anglickou klávesnicí), existuje program, který to udělá za nás. Jmenuje se *vlna* nebo *vlna32*⁴. Program *vlna32* (který budeme spíše volit ve Windows) se používá takto:

```
vlna32.exe soubor.tex
```

kde *soubor.tex* je zpracovávaný soubor. Dále je nutné přidat cestu k programu i zpracovávanému souboru.

Standardně bohužel program nezpracovává úplně vše, co by měl, ale máme možnost určovat, za která jednopísmenná slova bude vlnka přidána:

```
vlna32.exe -v KkSsVvZz0oUuAaIi soubor.tex
```

Tento způsob používání se může zdát poněkud pracný, ale pokud pracujeme v editoru *T_EXnicCenter* nebo jiném překládajícím v profilech, dá se trochu zautomatizovat. Následující postup platí pro *T_EXnicCenter*, v jiných editorech bude obdobný:

- v menu editoru najdeme okno Profiles (Build → Define output profiles),
- v okně vlevo dole klepneme na tlačítko Add,
- napíšeme název profilu (třeba *vlna*) a potvrdíme,
- v pravé polovině okna Profiles zaškrtneme políčka Do not use BibTeX... a Do not use MakeIndex... (odtud se nic spouštět nebude) a přepneme se na druhou záložku (Postprocessor),
- tam přidáme nový program (vlevo nahoře tlačítko se čtverečkem), napíšeme jeho název (je jedno jaký, třeba *vlna*) a do řádků níže zadáme:
 - doplníme nebo najdeme (tlačítko  vedle pole Path) adresu k programu *vlna32.exe*,
 - doplníme argumenty příkazu (Command line arguments)

```
-v KkSsVvZz0oUuAaIi "%pm"
```

(to znamená, že se bude vlnka doplňovat za uvedená samotná písmena a *%pm* se nahradí názvem toho souboru, se kterým právě pracujeme).

Pak stačí vždy jen zvolit tento profil v rozbalovacím seznamu na hlavním panelu a klepnout na tlačítko pro překlad.

⁴Program si můžeme stáhnout na [ftp://math.feld.cvut.cz/pub/olsak/vlna/](http://math.feld.cvut.cz/pub/olsak/vlna/). Potřebujeme soubor *vlna.exe* nebo *vlna32.exe* a dále soubor *vlna.txt*, kde je popsáno, jak se tento soubor používá.

2.5 Odstavcové styly

2.5.1 Prostředí

Prostředí je úsek textu, který má být určitým způsobem zformátován. Až na výjimky vyznačujeme začátek prostředí příkazem `\begin{název prostředí}` a konec příkazem `\end{název prostředí}`. Každé prostředí je zároveň skupinou (viz kap. 2.4, str. 44), tedy příkazy s platností do konce skupiny (třeba pro změnu stupně písma) končí svou platnost nejen u nejbližšího výskytu `}`, ale také `\end{...}`.

V T_EXu se běžně používá mnoho různých prostředí – například `center` pro zarovnání úseku textu na střed, `quotation` pro sazbu citátů, `itemize` jako výčtové prostředí s odrážkami, máme také několik matematických a tabulkových prostředí i prostředí, ve kterých se příkazy nevyhodnocují a považují se za běžný text. Některými z nich se budeme zabývat v následujícím textu, informace o dalších najdeme například v [T31], [T36], [T37] a dalších.

2.5.2 Formát odstavce

Parametry odstavce nastavujeme (většinou v preambuli – hlavičce – dokumentu) příkazy podle tabulky 2.3⁵. Pro vertikální odsazení odstavců a odstavcovou zarážku používáme násobky délek `ex` a `em`. Jsou odvozeny z rozměrů kuželky momentálního stupně písma, proto po změně stupně písma nebo fontu se optické vlastnosti odstavce příliš nezmění.

Ukázka příkazu	Význam
<code>\setlength{\parskip}{1ex}</code> ⁶	nastaví vertikální odsazení odstavců na půl délky <code>ex</code>
<code>\setlength{\parindent}{2em}</code>	nastaví odsazení prvního řádku odstavce (zarážky) na 2 <code>em</code>
<code>\def\baselinestretch{1.5}\normalsize</code>	nastaví řádkování na 1,5 násobek běžného řádkování

Tabulka 2.3: Nastavení parametrů odstavce

Odstavce oddělujeme prázdným řádkem nebo příkazem `\par`, první možnost je samozřejmě jednodušší. Dále máme možnost přerušit řádek kdekoliv uprostřed odstavce příkazem `\\` nebo `\newline` tak, že na novém řádku nebude použito žádné odsazení (prostě se jen přesuneme na nový řádek). Zároveň můžeme i vynechat místo, velikost mezery udáváme v nepovinném parametru – například `\\[2em]`. Pro ukončení stránky použijeme příkaz `\newpage`.

Běžný text je zarovnán do bloku, tedy levý i pravý okraj jsou rovné. To máme možnost změnit a nastavit zarovnání na střed nebo na praporek – praporek vlající

⁵V prvních dvou případech jde o práci s tzv. *délkovými registry* (délkami), jakýmiis zástupnými názvy pro často používané délky. V registru `\parskip` je uložena vzdálenost mezi odstavci, v registru `\parindent` velikost odstavcové zarážky.

⁶Zde je vhodné dát určitou rezervu, aby T_EX při vkládání konce stránky měl možnost mírně „pružně“ upravit vzdálenosti mezi odstavci. Pokud to neuděláme, T_EX si ji stanoví sám tak, že se nám to nemusí líbit. Příkaz může vypadat například takto: `\setlength{\parskip}{1ex plus .3ex minus .2ex}`.

vpravo („zubatý“ text vpravo, rovný vlevo) je zarovnání vlevo, praporek vlající vlevo je zarovnání vpravo.

Zarovnání	Příkaz
na střed	<code>\begin{center} ... \end{center}</code> <code>{ ... \centering ... }</code>
vlevo (praporek doprava)	<code>\begin{flushleft} ... \end{flushleft}</code> <code>{ ... \raggedright ... }</code>
vpravo (praporek doleva)	<code>\begin{flushright} ... \end{flushright}</code> <code>{ ... \raggedleft ... }</code>

Tabulka 2.4: Možnosti zarovnání textu

V tabulce 2.4 máme u každé možnosti dva způsoby, jak určitého zarovnání dosáhnout. První je pomocí prostředí, druhá příkazem platným do konce skupiny (ve skutečnosti platí už od začátku odstavce, ve kterém se nachází). Většinou je lepší použít prostředí, druhý způsob je vhodný třeba při umísťování obrázku nebo tabulky. Když nechceme psát příkazy, můžeme využít menu nebo hlavní panel editoru, kde najdeme tlačítka podobná těm z Wordu.

Výpis č. 2.7

Zatím ještě máme odstavce zarovnané do bloku, ale to brzy změním hned po ukončení řádku uvnitř odstavce.

`\begin{flushright}`
Toto je ukázka zarovnání textu k-pravému okraji. Platí až do konce prostředí, jestliže zarovnání nezměníme některým příkazem.

Tento `\centering` text je centrován, a to již od začátku odstavce.
`\end{flushright}`

Vracíme se k-zarovnání do bloku, předchozí prostředí už skončilo.

Zatím ještě máme odstavce zarovnané do bloku, ale to brzy změním, hned po ukončení řádku uvnitř odstavce.

Toto je ukázka zarovnání textu k pravému okraji. Platí až do konce prostředí, pokud zarovnání nezměníme některým příkazem.

Tento text je centrován, a to již od začátku odstavce.

Vracíme se k zarovnání do bloku, předchozí prostředí už skončilo.

Pokud je určeno nenulové odsazení prvního řádku (tzv. *odstavcová zarážka*), můžeme používání této zarážky řídit příkazy `\noindent` (pro tento odstavec se nepoužije) a `\indent` (pro tento odstavec se použije). Druhý příkaz je užitečný například tehdy, když se zarážka používá pro všechny odstavce kromě prvního v kapitole, a my vyjimečně chceme odsadit i ten první. Oba příkazy píšeme těsně před začátek odstavce, pro který mají platit:

`\noindent`

text odstavce, který nemá mít odsazený první řádek

Odstavce mohou být *pojmenované*. Takto lze nenásilně bez nadpisů členit příliš dlouhou kapitolu, ve které se určité odstavce týkají vždy určitého tématu. Název

odstavce je formátován tučně stejně jako názvy kapitol, to ale můžeme změnit použitím příkazu `\normalfont` a příkazu pro vybraný formát.

Výpis č. 2.8

```
\paragraph{Pojmenovaný odstavec.}
Pojmenované odstavce používáme pro
podrobnější členění kapitoly, kde
odstavce lze pojmenovat jejich
tématem. Všimněte si vyznačení
pojmenování a odsazení textu.
```

Někdy odstavec nelze pojmenovat -- pak ho necháme tak, jak je. Obvykle se počítá s tím, že tematicky patří k předchozímu odstavci.

```
\paragraph{Nadpisy.}
Nadpisy obvykle členíme do úrovní, ale maximálně čtyř,
ideálně tří. Pro podrobnější členění
můžeme použít pojmenované odstavce.
```

Pojmenovaný odstavec. Pojmenované odstavce používáme pro podrobnější členění kapitoly, kde odstavce lze pojmenovat jejich tématem. Všimněte si vyznačení pojmenování a odsazení textu.

Někdy odstavec nelze pojmenovat – pak ho necháme tak, jak je. Obvykle se počítá s tím, že tematicky patří k předchozímu odstavci.

Nadpisy. Nadpisy obvykle členíme do úrovní, ale maximálně čtyř, ideálně tří. Pro podrobnější členění můžeme použít pojmenované odstavce.

2.5.3 Seznamy

V T_EXu používáme tři druhy výčtových seznamů – výčtové (*itemize*), číslované (*enumeration*) a s popisky (s pojmenovanými položkami, *description*). Všechny tři druhy jsou tvořeny prostředím a položky začínají příkazem `\item`. U všech tří druhů můžeme v nepovinném parametru položky určit „popisek“, který však využijeme obvykle pouze u seznamu s popisky (to budou ty popisky), jinak tímto způsobem lze například změnit symbol pro odrážku.

Strukturu seznamu vytvoříme buď ručně podle příkladu 2.4 nebo pomocí tlačítek na hlavním panelu či položkami v menu (Insert → Enumerations), položky seznamu vložíme tamtéž (. . . Entry). V jednom prvku seznamu může být více odstavců.

V příkladu 2.4 je ve výpisu 2.9 ukázán způsob změny symbolu pro odrážku (v tomto smyslu však moc nepoužíváme), ve výpisu 2.10 vidíme možnost odkazování se na jednotlivé položky číslovaného seznamu (je to použitelné i mimo seznam), ve výpisu 2.11 je pozměněn formát posledního popisku.

Příklad 2.4 Ve třech výpisech vidíme ukázkou použití základních druhů seznamů.

Výpis č. 2.9

```
Výčtový seznam:
\begin{itemize}
\item první odrážka,
\item druhá odrážka,
\item[\CheckedBox] třetí odrážka.
\end{itemize}
```

Výčtový seznam:

- první odrážka,
- druhá odrážka,
- ☑ třetí odrážka.

Výpis č. 2.10

Vztahy mezi zvířaty,
odkaz na `\ref{seznam:kocka}`:
`\begin{enumerate}`
`\item` Pes.
`\item\label{seznam:kocka}` Kočka.
`\item` Myš utíká před tím, co je v~položce
`\ref{seznam:kocka}`.
`\end{enumerate}`

Vztahy mezi zvířaty, odkaz na **2**:

1. Pes.
2. Kočka.
3. Myš utíká před tím, co je v položce **2**.

Výpis č. 2.11

Seznam s~popisky:
`\begin{description}`
`\item`[Spojovník] píšeme jednoduše klávesou
s~tímto znakem.
`\item`[Pomlčka] se zadává jako dva
spojovníky.
`\item[\normalfont\itshape` Nezlomitelná
mezera] se nahrazuje vlnovkou.
`\end{description}`

Seznam s popisky:

Spojovník píšeme jednoduše klávesou s tímto znakem.

Pomlčka se zadává jako dva spojovníky.

Nezlomitelná mezera se nahrazuje vlnovkou.

Výpis č. 2.12

Víceúrovňové číslování:
`\begin{enumerate}`
`\item` Toto je první úroveň, první odstavec.

Tady je druhý odstavec.
`\item` Další položka první úrovně.
`\begin{enumerate}`
`\item` Toto je druhá úroveň.
`\item` Další položka.
`\end{enumerate}`
`\end{enumerate}`

Víceúrovňové číslování:

1. Toto je první úroveň, první odstavec.
Tady je druhý odstavec.
2. Další položka první úrovně.
 - (a) Toto je druhá úroveň.
 - (b) Další položka.

Výpis 2.12 je ukázkou možnosti vnořování výčtových prostředí, čímž vzniká třeba víceúrovňové číslování. Seznamy můžeme libovolně kombinovat včetně různých typů, $\LaTeX$ se vždy sám postará o vhodné odlišení úrovní. Jen si musíme dávat pozor na „překřížení“ začátků a konců různých prostředí.

Vlastnosti prostředí pro seznamy se určují obvykle hned za úvodním příkazem prostředí. Například vertikální mezery mezi položkami seznamu změníme takto:

```
\begin{itemize}\setlength{\itemsep}{0pt} \item ...
```

U seznamů se dále mohou hodit délkové registry `\partopsep`, `\labelsep`, `\itemindent`, `\leftmargini`, `\leftmarginii`, `\leftmarginiii`, `\leftmarginiv`.

2.6 Dokumenty

2.6.1 Třída dokumentu

Typ (třidu) dokumentu určujeme v příkazu `\documentclass`. Standardní třídy jsou:

article – podle názvu by se mohlo zdát, že je použitelná pouze pro články, ale je to nejpoužívanější třída pro běžné texty do velikosti několika desítek stran. Výchozí nastavení je jednostranný dokument.

report – třída vhodná pro dokumenty v délce několika desítek až stovek stran, výchozí nastavení je jednostranný dokument.

book – třída vhodná pro rozsáhlé dokumenty. Oproti třídě *report* je nastaven oboustranný dokument a jinak zformátováno záhlaví a zápatí.

letter – třída pro psaní dopisů.

Nejčastěji budeme používat třídu *article*.

Zároveň s určením třídy obvykle nastavujeme některé vlastnosti, které se nám ve výchozím nastavení nelíbí. Tyto vlastnosti se uvádějí do nepovinného parametru. Například když chceme použít třídu *article*, ale nastavit rozměr stránky A4, stupeň písma 12 a oboustranný dokument, uděláme to takto:

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

Nikde neděláme mezery, vlastnosti mohou být v jakémkoliv pořadí.

Kromě standardních tříd existují ještě další třídy, obvykle vytvořené lidmi k určitému konkrétnímu účelu. Hodně tříd je pro vytváření prezentací (například *beamer*, *prosper*, *ifmslide*, *texpower*, *elpres*), zasilání článků do odborných časopisů (*paper*, *llncs*, *elsevier*, *acmtrans2e*, *amsart*, *nature*), diplomové a disertační práce (*scrreprt*, *scrartcl*, *adfathesis*, *classicthesis*, *ua-thesis*), referenční manuály (*refart*, *refrep*), odborné knihy a beletrie (*scrreprt*, *scrbook*, *hitec*, *memoir*, *octavo*, *amsbook*), životopisy (*cv*, *moderncv*), letáky (*leaflet*), laboratorní záznamy (*labbook*), postery (plakáty většinou v rozměru A0, používají se například na konferencích pro grafickou prezentaci výsledků výzkumu, máme možnost určovat pozici jednotlivých objektů či textu na stránce) (*a0poster*) atd.

Seznamy z předchozího odstavce rozhodně nejsou kompletní. Počet těchto tříd je různý v různých distribucích T_EXu, je jich však více, než běžný uživatel tuší (běžnému uživateli ostatně stačí standardní třídy). Později, v kapitole 2.6.2, si ukážeme, jak o nich najít informace (tyto třídy se liší nejen nastavením stylů, tedy „vizuálně“, ale někdy i způsobem použití – definovanými příkazy) a jak je využít v dokumentech.

2.6.2 Styly

Podobně jako jiné systémy, i L^AT_EX umožňuje ovlivňovat vzhled dokumentu úpravou stylů. Pro jednodušší dokumenty dostačují předdefinované styly (pro odstavce, nadpisy apod.), tyto styly lze ovlivňovat následovně:

1. *Výběrem třídy dokumentu.* Každý dokument má přiřazenu právě jednu třídu, která stanoví základní vzhled a strukturu a případně přidává některé specifické

příkazy, prostředí a objekty k té třídě, ze které je odvozena. S některými třídami jsme se seznámili v kapitole 2.6.1, tedy už víme, že třída se načítá příkazem

`\documentclass[nepovinné argumenty]{název třídy}`

Například `\documentclass[a4paper,11pt,twoside]{report}` znamená, že pro dokument chceme použít třídu `report`, budeme tisknout na papír formátu A4, velikost písma je 11 bodů a jde o oboustranný dokument.

2. *Načtením balíčku.* V balíčku (package) je nadefinován nebo pozměněn jen jeden nebo více stylů nebo je vytvořen nový příkaz či prostředí pro některý konkrétní účel, obvykle lze balíček používat zároveň s jinými balíčky nebo kteroukoliv třídou. Už dříve jsme používali příkaz pro načtení balíčku

`\usepackage[nepovinné argumenty]{název balíčku}`

Nepovinné argumenty jsme u balíčků ještě nepoužívali, seznámíme se s nimi například u balíčku `hyperref` nebo `color`.

V jednom příkazu `\usepackage` můžeme načíst více balíčků, jak uvidíme v příkladu 2.5.

3. *Změna nebo vytvoření stylu přímo v dokumentu.* V dokumentu, obvykle v jeho hlavičce (preambuli), si můžeme vytvořit vlastní příkazy nebo prostředí nebo pozměnit existující. Ve většině případů používáme příkazy `\def`, `\newcommand`, `\renewcommand`, `\newenvironment`, `\renewenvironment` nebo pro nastavení délek `\setlength`. Pro číslované prostředí je výhodné použít třeba příkaz `\newtheorem` v kombinaci s jiným příkazem. Ukázky použití těchto příkazů jsou ve výpisu 2.14 v příkladu 2.5.
4. *Jiné.* Některé styly se načítají trochu jiným způsobem. Jde například o různé typy a řezy písma. My jsme si v kapitole 2.4.3 o fontech ukazovali změnu typu písma načtením balíčku, ale obvyklejší (a o něco složitější) cesta je pomocí dalších příkazů (viz [T31]). Dále některé zvláštní řezy písma se načítají trochu netradičně, například řez *tapír* se používá takto:

Výpis č. 2.13

```
% inicializace třeba v preambuli dokumentu:
\font\tapir tap

% používání kdekoliv v dokumentu:
{\tapir Tento text je psán řezem tapír.\par Bohužel
v~něm nefungují kombinace s~dalšími řezy.}
```

Tento text je psán řezem tapír.

Bohužel v něm nefungují kombinace s dalšími řezy.

Příklad 2.5 Ve výpisu máme ukázkou jednoduché preambule dokumentu typu článek (article) s fontem `bookman` a také vidíme způsob použití výše uvedených příkazů pro definování nových příkazů či stylů. Přímou z příkladů lze odvodit způsob používání jednotlivých příkazů včetně potřebných argumentů.

Výpis č. 2.14

```

\documentclass[a4paper,12pt]{article}
\usepackage{czech,a4wide,comment} % nikde v parametrech nesmí být mezera!!!
% balíček a4wide zajistí obvyklé rozměry okrajů stránky pro formát A4,
% balíček comment umožní používat prostředí comment pro víceřádkové komentáře
% \begin{comment} ... \end{comment}
\usepackage{bookman}

\usepackage{textcomp,gensymb,marvosym,wasysym} % balíčky s různými symboly

% příkaz \zhlavitabulky se použije pro styl záhlaví tabulky, na tomto místě
% (v preambuli dokumentu) to lze jednoduše změnit pro celý dokument:
\def\zhlavitabulky{\bfseries}

% vytváříme vlastní příkaz s jedním parametrem:
\newcommand{\kolecko[1]{\textcircled{\tiny #1}}
% používá se například takto: \kolecko{10} - vytvoří číslo 10 v kolečku

% vytvoříme jednoduché prostředí mírně pozměňující původní prostředí itemize,
% pozměníme mezery mezi položkami seznamu (všimněte si strukturování příkazu
% pomocí komentářů):
\newenvironment{mitemize}%
{ \begin{itemize} \setlength{\itemsep}{0pt} }%
{ \end{itemize} }
% používá se: \begin{mitemize} ... \end{mitemize}

% následující prostředí slouží k vhodnému formátování poznámek - na začátku
% slovo Poznámka následované mezerou a na konci podtržení celého odstavce:
\newenvironment{poznámka}%
{ \textbf{Poznámka:} \quad }%
{ \newline \rule{0pt}{1pt} \hrulefill \rule{0pt}{1pt} }

% potřebujeme číslované prostředí příklad, číslování bude záviset na číslu
% kapitoly, využijeme kombinaci newtheorem a newenvironment:
\newtheorem{thpříklad}{Příklad}[chapter]
\newenvironment{příklad}%
{ \begin{thpříklad} \rule{1pt}{0pt} \normalfont }%
{ \end{thpříklad} }
% používá se: \begin{příklad} ... \end{příklad}

```

U příkazů `\def` a `\newcommand` jsou argumenty celkem jasné – je potřeba pouze zadat, co se stane, když je vytvářený příkaz použit, v nepovinném parametru můžeme určit případné parametry příkazu (pro `\def` viz nápověda).

U příkazu `\newenvironment`, který vytvoří nové prostředí, zadáváme po jeho názvu nejdřív, co se má provést při `\begin{nové prostředí}`, tedy na jeho začátku, a v dalším argumentu určíme, co se má provést při `\end{nové prostředí}`. U prostředí definovaného příkazem `\newtheorem` obvykle pouze určíme jeho název a vazbu číslování (máme více možností, než je zde naznačeno, podrobnosti najdeme v nápovědě).

Častou a nenápadnou chybou je „chaos“ v závorkách. Když nám program hlásí při překladu chybu související se závorkami nebo s hranicemi prostředí, nezbyvá nic jiného než počítat levé a pravé závorky a kontrolovat jejich umístění. Zde se vyplácí psát přehledně jak je to ukázáno v předchozím příkladu.

Třídy jsou definovány v souborech s příponou `.cls`, balíčky v souborech s příponou `.sty`. Najdeme je obvykle v adresáři, kde je $\text{\TeX}$ nainstalován, a to v podadresáři `... \tex\latex\název třídy` či balíčku. Dokumentace k těmto souborům bývá pak v adresáři `... \doc\latex\název třídy` či balíčku, většinou v `.dvi` nebo `.pdf` souborech, někdy doprovázená soubory s příklady.

Na internetu (na [P19]) lze někdy sehnat novou třídu nebo balíček, případně novou verzi. Při instalaci postupujeme podle pokynů v souboru dodávaném s třídou nebo balíčkem, obvykle jde jen o to v adresáři `... \tex\latex` vytvořit vhodně pojmenovaný adresář, do něho zkopírovat několik souborů a pak obnovit databázi.

O jakou databázi vlastně jde? $\text{\TeX}$ se skládá ze spousty velmi malých souborů. Aby ho časté přistupování k těmto souborům zbytečně nezdržovalo (především jejich hledání na disku), udržuje si jejich databázi, kde má uloženy veškeré potřebné informace, které mohou urychlit přístup k používaným souborům. To značně zrychluje práci, ale vedlejším efektem je, že když provedeme změny ve struktuře těchto souborů (včetně přidání nového balíčku), musíme o tom $\text{\TeX}$ u „dát vědět“. U $\text{\MiKTeX}$ u ve Windows se to dělá v menu  pomocí  $\rightarrow$ $\text{\MiKTeX}$ $\rightarrow$ $\text{\MiKTeX Options}$, tlačítko `Refresh Now`.

Úlohy:

1. Zjistěte, ve kterém adresáři (složce) je na vašem počítači nainstalován $\text{\TeX}$, a kde jsou soubory s třídami, balíčky a jejich dokumentací. Nezapomeňte, že samotný operační systém se svou funkcí hledání vám může ulehčit práci.
2. Srovnajte dokumentace k třídám `jasthesis`, `adfathesis`, `scrreprt` (je v adresáři `koma-script`, v souboru `scrguien.pdf`) a `octavo`. Všechny tyto třídy vyzkoušejte na jednoduchém dokumentu s nadpisy a textem.
3. Najděte dokumentaci k těmto balíčkům:
 - (a) balíček `clock`, zjistěte, jak se vkládá obrázek hodin a jak se ovládá jeho vzhled,
 - (b) balíčky pro úpravu nadpisů – `fncychap`, `sectsty`, `quotchap`,
 - (c) balíček pro tvarování odstavců – `shapepar`.
4. Zjistěte, k čemu slouží balíčky `parallel` a `qsm`.
5. Najděte dokumentaci balíčků pro prezentace `beamer` a `prosper`, u obou se podívejte na příklady (u `prospera` bude nutné provést překlad $\text{\LaTeX} \Rightarrow \text{\PS} \Rightarrow \text{\PDF}$).

2.6.3 Nadpisy

Na tom, pro jakou třídu dokumentu jsme se rozhodli, závisí nejen vzhled výsledného dokumentu a výchozí nastavení vlastností (jednostranný, oboustranný apod.), ale v některých případech i příkazy, které máme k dispozici. Týká se to především nadpisů.

Ve výpisu 2.15 je způsob vkládání nadpisů prvních tří úrovní, a to v prvním sloupci pro třídu `article`, v druhém pro třídy `report` a `book`. Jak vidíme, `article` nepoužívá vůbec příkaz `\chapter`. Tento příkaz totiž generuje graficky řešené, „zdobnější“ a nápadnější nadpisy první úrovně, kde obvykle bývá nadpis samotný na jiném řádku než číslo kapitoly, zatímco příkazy obsahující slovo `section` generují jednořádkové nadpisy s číslem a textem nadpisu.

Výpis č. 2.15

<code>\section{Nadpis první úrovně}</code>	<code>\chapter{Nadpis první úrovně}</code>
Text první kapitoly článku.	Text první kapitoly dokumentu.
<code>\subsection{Nadpis druhé úrovně}</code>	<code>\section{Nadpis druhé úrovně}</code>
Text podkapitoly.	Text podkapitoly.
<code>\subsubsection{Nadpis třetí úrovně}</code>	<code>\subsection{Nadpis třetí úrovně}</code>
atd.	atd.

Někdy je třeba vložit *nečíslovaný nadpis*, ať už kterékoliv úrovně (třeba pro úvodní kapitolu). To provedeme mírnou změnou příkazu pro vložení nadpisu, a to přidáním hvězdičky. Pro různé úrovně tedy použijeme příkazy:

`\chapter*{Nadpis první úrovně v report a book}`

`\section*{Nadpis druhé úrovně v report a book nebo první v article}`

atd. pro další úrovně. Nečíslované nadpisy se však nezahrnou do obsahu, ale i to lze řešit přidáním příkazu pro přidání položky obsahu – podle příkladu 2.6.

Příklad 2.6 Úvodní kapitola má být nečíslovaná, ale chceme ji v obsahu dokumentu. Vlevo je řešení pro třídu `article`, vpravo pro `report` a `book`:

Výpis č. 2.16

<code>\section*{Úvod}</code>	<code>\chapter*{Úvod}</code>
<code>\addcontentsline{toc}{section}{Úvod}</code>	<code>\addcontentsline{toc}{chapter}{Úvod}</code>
text kapitoly článku	text kapitoly rozsáhlejšího dokumentu

V parametrech příkazu zkratka `toc` znamená *Table of Contents*, tedy obsah. Když použijeme `lof`, vkládáme do seznamu obrázků (*List of Figures*), když použijeme `lot`, vkládáme do seznamu tabulek (*List of Tables*). Dalšími možnostmi ovlivňování dokumentových seznamů se budeme zabývat v kapitole 2.12.

Dokument může být členěn také na větší díly než kapitoly. Když chceme shrnout více kapitol do většího celku, části, pak pro jednotlivé části použijeme „nadnadpis“ `\part{Název části}`.

Pro oddělení příloh od běžných kapitol použijeme příkaz `\appendix`. Od tohoto příkazu dále budou nadpisy nejvyšší úrovně číslovány velkými písmeny od „A“.

Ve třídě `book` jsou k dispozici příkazy, které rozčlení dokument na části lišící se především číslováním stránek – `\frontmatter`, `\mainmatter` a `\backmatter`. Způsob jejich použití je naznačen ve výpisu 2.17.

Výpis č. 2.17

```
...
\begin{document}
\frontmatter % přední část dokumentu
... titulní strany, předmluva, obsah, seznam tabulek, apod.
\mainmatter % hlavní část dokumentu
... běžné kapitoly dokumentu, číslované číslly
\appendix
... přílohy číslované písmeny, také patří do hlavní části dokumentu
\backmatter % koncová část dokumentu
... seznam literatury, rejstřík apod.
```

2.6.4 Titul a abstrakt

Už ve výpisu 2.1 na straně 39 jsme si mohli všimnout, že článek nemá jen název, ale je uveden i autor, datum a abstrakt.

Název dokumentu, seznam autorů a datum jsou součástí titulu, některé typy dokumentů do titulu řadí i jiné informace. Zatímco v případě článku (třída `article`) je titul jen jednou ze součástí první strany, třídy `report` a `book` vytvářejí automaticky celou titulní stranu. Většinou používáme následující sekvenci příkazů:

Výpis č. 2.18

```
\title{Název dokumentu}
\author{Autor dokumentu} % Více autorů oddělíme příkazem \and
\date{\today} % Když příkaz vynecháme, automaticky se doplní skutečné datum,
               % pokud příkaz použijeme, ale prázdný, datum nebude součástí
               % titulu vůbec

\maketitle
```

V editoru `TeXnicCenter` nemusíme tyto příkazy vypisovat, můžeme použít menu `Insert` → `Document Title`, kde `Title Properties` vloží příkazy pro název, autora a datum, `Title` pak příkaz `\maketitle`.

Volbu `Title Page` využijeme, pokud chceme i ve třídě `article` mít celou titulní stranu. Je to prostředí, které má obsahovat příkazy vložené volbou `Title Properties` a nahrazuje příkaz `\maketitle`.

Do parametrů uvedených příkazů nemusíme vkládat pouze text, můžeme použít příkazy pro jinou velikost písma, vkládat více řádků, použít různé řezy písma či dokonce barvy a obrázky (u těch je trochu problém s umístěním).

V menu `Insert` → `Document Title` najdeme i volbu `Abstract` k vložení *abstraktu* (nebo napíšeme `\begin{abstract} ... \end{abstract}`). Ten následuje za titulem.

Když píšeme článek nebo dokument s použitím jiné než standardní třídy, obvykle s touto třídou máme k dispozici jakýsi „návod k použití“, což je dokument psaný s použitím této třídy a popisuje její možnosti a způsob, jak je využívat. V kapitole 2.6.2 je o této problematice více.

2.6.5 Poznámky pod čarou a na okraj

Občas potřebujeme vložit do textu *poznámku pod čarou*. Ta se vkládá příkazem

```
\footnote{Text poznámky pod čarou.}
```

Před tímto příkazem obvykle nebývá mezera (znak poznámky pod čarou musí přiléhat k tomu, co komentujeme), v poznámce píšeme „celými větami“, tedy ještě před ukončující závorkou bývá tečka či jiné interpunkční znaménko, a to i v případě, že je i za závorkou.

Poznámka na okraj.

Poznámka na okraj se vytvoří příkazem

```
\marginpar[text pro levý okraj]{text pro (pravý) okraj}
```

Text se vloží jako poznámka k pravému okraji textu u jednostranného dokumentu, u oboustranného vždy na vnější stranu. Pokud zadáme i nepovinný parametr, tak v případě, že u oboustranného dokumentu vyjde poznámka vlevo, použije se text (případně s příkazy) z nepovinného parametru.

Poznámka je vždy zarovnána ke svému levému okraji, ale nic nám nebrání použít příkaz nebo prostředí pro zarovnání doprava na levé stránce, jak je tomu i ve výpisu 2.19 (poznámka vytvořená druhým odstavcem výpisu je u předchozího textu), případně různé formátovací příkazy.

Výpis č. 2.19

Poznámky pod čarou můžeme vložit téměř kdekoliv, například tady `\footnote{Toto je poznámka pod čarou.}`, místo je označeno obvykle číslem. Poznámky na okraj `\marginpar{\raggedleft Toto je poznámka na okraj.}{Toto je poznámka na okraj.}` se objeví na okraji v-linii umístění tohoto příkazu.



V poznámce na okraj nemusí být jenom text, ale také obrázek nebo text různě upravený, například `\marginpar{\raisebox{-1em}{\fbox{\huge\bfseries ?}}}`.

Poznámku na okraj nelze přímo použít uvnitř některých prostředí, například v tabulkách nebo v prostředí `minipage`. Existuje však možnost, jak to obejít (je použita například v tabulce 2.3 na straně 52) – na místě, kde chceme poznámku vložit, napíšeme místo příkazu `\footnote` příkaz

```
\footnotemark
```

a hned za koncem daného prostředí (třeba tabulky nebo `minipage`) použijeme příkaz

```
\footnotetext{Text poznámky pod čarou.}
```

2.6.6 Záhloví a zápatí

Stránky dokumentu jsou obvykle číslovány, a pokud je vytvořena titulní strana, automaticky není číslována. Způsob číslování můžeme ovlivnit různými způsoby. Jeden z nich jsme si už řekli – ve třídě `book` pomocí `\frontmatter`, `\mainmatter` a `\backmatter`.

Styl číslování úzce souvisí s definováním záhlaví a zápatí. Ve třídách `article` a `report` je číslo stránky vždy v zápatí uprostřed, záhlaví je prázdné, ve třídě `book` je v záhlaví číslo stránky a název kapitoly a na stránkách, kde začíná hlavní kapitola, je záhlaví prázdné a číslo stránky je v zápatí uprostřed.

Styl stránky včetně umístění číslování ovlivníme příkazy

```
\pagestyle{styl stránky}
\thispagestyle{styl stránky}
```

kde styl stránky může standardně být

- `empty` – záhlaví a zápatí jsou prázdné, stránky nejsou číslovány,
- `plain` – záhlaví je prázdné a v zápatí uprostřed je číslo stránky – výchozí pro třídy `article` a `report`,
- `headings` – platí to, co bylo popsáno pro třídu `book`,
- `myheadings` – můžeme do určité míry určovat obsah záhlaví a zápatí.

První příkaz bývá často v preambuli, ale může být kdekoli v dokumentu, platí od stránky, na které je uveden. Druhý příkaz platí pouze pro stránku, na které je uveden. Tyto příkazy nemají vliv na hodnotu čísla stránky, jen na to, zda a kde je.

Pokud je nastaven styl `myheadings`, máme k dispozici příkazy pro změnu obsahu záhlaví:

```
\markright{pravá hlavička}
\markboth{levá hlavička}{pravá hlavička}
```

První příkaz se používá pro jednostranné dokumenty, druhý pro oboustranné. Do parametru můžeme napsat jakýkoliv text (třeba název dokumentu).

U těchto příkazů je nutno si dát pozor na začátek jejich platnosti. Záhlaví na *levé* stránce je vygenerováno podle *posledního* použití příkazu `\markboth` před začátkem této stránky, zatímco záhlaví na *pravé* stránce podle *prvního* použití `\markboth` nebo `\markright` na této stránce.

Další příkaz, tentokrát ovlivňující i hodnotu čísla, je

```
\pagenumbering{typ číslic}
```

Typ číslic může být `arabic` (běžné arabské číslice), `roman` (malé římské číslice) nebo `Roman` (velké římské číslice).

Tento příkaz, jak je z parametru patrné, mění typ číslic pro číslo stránky, platí od strany, na které je příkaz uveden. Pokaždé je číslo stránky vynulováno, tedy na straně, kde je příkaz uveden, je vždy číslo 1, i nebo I (ne 0, protože číslo je vždy před vložením na stránku zvýšeno o 1).

Mnohem lepší možnosti ovlivnění tvaru záhlaví a zápatí máme s balíčkem `fancyhdr`. Používá se stejně jako předdefinované styly stran, tedy po načtení balíčku příkazem `\usepackage` použijeme příkaz `\pagestyle{fancy}` a dále definujeme záhlaví a zápatí příkazy z tabulky 2.5.

Parametr pozice u posledních dvou příkazů tabulky 2.5 se skládá ze dvou písmen. První určuje, zda se obsah má vložit vlevo (L), doprostřed (C) nebo vpravo

<i>Jednostranný dokument:</i>	
<code>\lhead{obsah}, \chead{obsah}, \rhead{obsah}</code>	levá, střední, pravá část záhlaví
<code>\lfoot{obsah}, \cfoot{obsah}, \rfoot{obsah}</code>	levá, střední, pravá část zápatí
<i>Oboustranný dokument:</i>	
<code>\fancyhead[pozice]{obsah}</code>	obsah záhlaví na dané pozici
<code>\fancyfoot[pozice]{obsah}</code>	obsah zápatí na dané pozici

Tabulka 2.5: Příkazy balíčku fancyhdr

(R) na dané stránce, druhý udává, jestli se má příkaz použít pro liché (E – even) nebo sudé (O – odd) stránky. Když chceme určit více různých dvojic, oddělíme je čárkou. Tedy pokud chceme mít na lichých stránkách v záhlaví vlevo název dokumentu a v zápatí na sudých stránkách vlevo a lichých vpravo datum, použijeme příkazy

```
\fancyhead[LE]{Název našeho dokumentu}
\fancyfoot[LO,RE]{\today}
```

V parametrech příkazů můžeme také použít příkaz, který vygeneruje název hlavní kapitoly (napíšeme `\leftmark`), kapitoly druhé úrovně (`\rightmark`) nebo číslo stránky (`\thepage`). Dále lze nastavit čáry oddělující záhlaví nebo zápatí od textové části stránky (jejich výšku):

```
\renewcommand{\headrulewidth}{tloušťka}
\renewcommand{\footrulewidth}{tloušťka}
```

Příklad 2.7 Ukážeme si vytvoření jednoduchého záhlaví a zápatí v jednostranném (vlevo) a oboustranném (vpravo) dokumentu pomocí balíčku fancyhdr.

Výpis č. 2.20

```
...
\pagestyle{empty}
... stránky bez číslování
\pagestyle{fancy}
\pagenumbering{Roman}
\cfoot{}
\rhead{\thepage}
\lhead{}
\renewcommand{\headrulewidth}{0pt}
... obsah; římské číslování stránek
\lhead{\leftmark}
\renewcommand{\headrulewidth}{.8pt}
\pagenumbering{arabic}
... stránky s arabským číslováním

...
\pagestyle{empty}
... stránky bez číslování
\pagestyle{fancy}
\pagenumbering{Roman}
\fancyfoot[CE,CO]{}
\fancyhead[LO,RE]{\thepage}
\fancyhead[LE,RO]{}
\renewcommand{\headrulewidth}{0pt}
... obsah; římské číslování stránek
\fancyhead[LE]{\leftmark}
\fancyhead[RO]{\rightmark}
\renewcommand{\headrulewidth}{.8pt}
\pagenumbering{arabic}
... stránky s arabským číslováním
```

2.6.7 Dlouhé dokumenty

Samotnému $\text{\TeX}$, $\text{\LaTeX}$ a ani programům zobrazujícím DVI, PS a PDF soubory obvykle dlouhé dokumenty nečiní žádné problémy. Přesto však především pro přehlednost, snadnější orientaci a také pro zjednodušení práce s editorem je vhodné velmi dlouhý zdrojový soubor rozdělit do více souborů.

Všechny soubory, které takto vytvoříme, budou mít příponu TEX a měly by být ve stejném adresáři (složce). Jeden ze souborů bude *hlavní*. V něm je dána struktura celého dokumentu včetně preambule, titulních stran, obsahu apod., tedy styl dokumentu určujeme právě zde. Tento soubor pojmenujeme tak, jak chceme mít pojmenovaný i výsledný dokument (kromě přípony, samozřejmě).

Jednotlivé části (třeba kapitoly) uložíme do vhodně pojmenovaných souborů s příponou TEX a vložíme do hlavního dokumentu pomocí některého z těchto příkazů:

```
\input{soubor}
\include{soubor}
```

V obou případech píšeme název souboru *bez přípony*. První z těchto dvou příkazů vlastně jen provede při každém překladu „vkopírování“ vkládaného souboru do toho, kde je příkaz uveden, druhý před a po tomto přenesení navíc vloží stránkový zlom (tedy se přejde na novou stránku). Při vkládání celých kapitol, které mají začínat na nové stránce, proto používáme příkaz `\include`, jindy příkaz `\input`. Když potřebujeme stránkový zlom před souborem, ale ne za ním, vložíme soubor příkazem `\input` a před ním použijeme třeba příkaz `\newpage`.

Příklad 2.8 Ve výpisu 2.21 vidíme možný tvar hlavního souboru `chaloupka.tex`. Vkládané soubory nemusí mít žádnou zvláštní strukturu, obvykle začínají příkazem `\chapter`.

Výpis č. 2.21

```
\documentclass[11pt,a4paper]{report}
\usepackage{czech,a4wide}

\title{Perníková chaloupka\[\[1em] Možný postup konstrukce}
\author{Jan Novák}
\date{\itshape Poslední aktualizace: \today}

\begin{document}
\maketitle
\tableofcontents % vložení obsahu

\include{uvod} % vložení souboru uvod.tex, nečíslovaná kapitola Úvod
\include{material} % vložení souboru material.tex s kapitolou Použitý materiál
\newpage\input{konstrukce1}
% soubor konstrukce1.tex s první částí kapitoly Postup konstrukce
\input{konstrukce2} % soubor konstrukce2.tex, druhá část kapitoly
\include{statika} % vložení souboru statika.tex, kapitola Statika objektu
\include{limitypouziti} % limitypouziti.tex, kapitola Limity použití konstrukce

\appendix % následují přílohy:
\include{prilohaopohadce} % prilohaopohadce.tex, první příloha
\include{konstrukcninakresy} % konstrukcninakresy.tex, druhá příloha
```

```

\include{zaver}    % vložen soubor zaver.tex, nečíslovaná kapitola Závěr
\include{literatura}
    % literatura.tex, nečíslovaná kapitola se seznamem literatury
\end{document}

```

Překládáme vždy jen hlavní dokument, tedy například v editoru T_EXnicCenter bychom se měli před každým překladem přesunout ze souboru, se kterým pracujeme (a také ho uložit), do hlavního souboru a pak teprve spustit překlad. Tento postup nám zjednodušuje možnost práce s *projekty*, kterou nabízí mnoho editorů určených pro práci s T_EXem a L^AT_EXem. Bohužel v každém editoru se s projekty pracuje trochu jinak, včetně přípon pomocných souborů projektu.

T_EXnicCenter ukládá informace o projektu do pomocných souborů pojmenovaných stejně jako hlavní soubor, a to s příponou TCP (nastavení editoru, jako je verze, jazyk, naposledy použitý profil pro překlad, název hlavního souboru apod.) a TPS (které soubory jsou právě otevřené–rozpracované, velikost a pozice oken těchto souborů, pozice kurzoru v souboru apod.). Oba soubory jsou textové, tedy je můžeme otevřít třeba v Poznámkovém bloku.

Do těchto souborů se ukládají data při každém uzavření projektu. Projekt otevřeme jednoduše tak, že poklepeme na soubor TCP a získáme stejné pracovní prostředí jako při posledním ukončení práce.

Překlad projektu se provádí jiným tlačítkem a klávesovou zkratkou než v případě jednoho dokumentu – používáme tlačítko  nebo klávesovou zkratku . Máme také k dispozici strukturu objektu, seznam souborů a seznam objektů v okně Navigator, které si zobrazíme přes menu View → Navigator Bar.

Nový projekt vytvoříme (v editoru T_EXnicCenter) přes menu File → New project a existující otevřeme přes File → Open project nebo v případě, že jsme s ním na tomto počítači už pracovali, rychlejší cestou přes File → Recent Projects. Může se také hodit možnost rychle bez hledání otevřít hlavní soubor projektu, pokud tak editor neučinil, a to přes Project → Open main file.

Projekt lze vytvořit i dodatečně, stačí v menu editoru zvolit Project → Create with active file as main file.

2.7 Vícesloupcová sazba

Někdy se může hodit možnost sazby do více sloupců, zúžení odstavce nebo dokonce naskládání několika takových odstavců vedle sebe (jeden z postupů je využit i v tomto dokumentu pro umístění výpisu zdrojového textu a výsledku vedle sebe).

Pro vícesloupcovou sazbu nebo skládání bloků textu třeba i vedle sebe jsou v T_EXu tyto možnosti:

1. Při určení třídy dokumentu použijeme také parametr twocolumn, například:

```

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

```

Pak se sice titul dokumentu vysází „jednosloupcově“, ale text dokumentu bude ve dvou sloupcích. Tento způsob není použitelný, pokud chceme v dokumentu měnit dvousloupcovou a jednosloupcovou sazbu nebo používat více než dva sloupce.

2. Použijeme balíček `multicol`, ve kterém je definováno prostředí `multicols`, tedy do hlavičky dokumentu umístíme příkaz pro načtení balíčku

```
\usepackage{multicol}
```

a text, který chceme zformátovat do více sloupců, uzavřeme do prostředí:

```
\begin{multicols}{počet sloupců} text \end{multicols}
```

Například pro tři sloupce:

```
\begin{multicols}{3}
text, který chceme mít ve více sloupcích
\end{multicols}
```

Všechny sloupce budou mít stejnou šířku.

3. Text nebo jiné objekty umístíme do příkazu `parbox`, který se používá následovně:

```
\parbox[zarovnání od předchozího]{šířka}{text nebo něco jiného}
```

Nepovinných parametrů je ve skutečnosti více než jeden (dokonce i výška `parboxu`), podrobnosti viz nápověda nebo [T31]. V nepovinném parametru, který zde máme uveden, můžeme zadat některé z písmen `t`, `b`, `c` (`top`, `bottom`, `center`) pro vertikální pozici celého boxu vzhledem k okolnímu textu.

4. Podobně se používá prostředí `minipage`:

```
\begin{minipage}[zarovnání od předchozího]{šířka}
text nebo něco jiného
\end{minipage}
```

Šířku můžeme udat v běžných jednotkách, ale protože se obvykle vztahuje k šířce stránky (tu vrací příkaz `\textwidth`), často se používá údaj, který se odvíjí právě od šířky stránky. Například pokud chceme zadat o něco méně než polovinu stránky, napíšeme jako šířku údaj `.49\textwidth` a automaticky se šířka `parboxu` nastaví na 0,49-násobek šířky stránky⁷.

Při skládání `parboxů` a prostředí `minipage` je můžeme sázet přímo vedle sebe nebo oddělovat jinými objekty (třeba příkazem `\rule` s některým parametrem nulovým, příkazem `\hfill` nebo jiným `parboxem`).

Příklad 2.9 Ukážeme si použití prostředí `multicols`, příkazu `parbox` a prostředí `minipage`. `parbox` a `minipage` ohraničíme rámečkem (příkaz `\fbox`), abychom viděli skutečnou zabranou šířku. Použijeme také příkaz `\rule`, který vykreslí obdélník

⁷Všimněte si, že mezi reálným číslem `.49` (tedy `0,49`) a příkazem `\textwidth` se sice symbol pro násobení nepíše, ale vyhodnocuje se zde.

(v prvním případě o nulové výšce a pak o výšce dva body), jeho účelem je oddálit od sebe předchozí a následující objekty o zadanou „šířku“. Druhý má výšku nenulovou, abychom viděli skutečnou pozici obdélníku.

Výpis č. 2.22

```
\begin{multicols}{2}
Toto je text, který bude zarovnán do dvou sloupců, protože v~parametru prostředí
\texttt{multicols} je číslo 2. Kdyby tam bylo třeba číslo 4, tak by byl text
zarovnaný do čtyř sloupců. V~tomto prostředí mohou být trochu problémy třeba
s~popisky obrázků nebo poznámkami pod čarou.
\end{multicols}

\noindent
\fbbox{    % všimněte si odsazení dalšího textu zleva - pro přehlednost
  \parbox{.45\textwidth}{
    Toto je obsah prvního boxu o šířce 0,45-násobku šířky textu na stránce.
  }
}
\rule{1ex}{0pt}    % tento obdélník nebude vidět
\fbbox{
  \parbox{.5\textwidth}{
    Toto je obsah druhého boxu širokého polovinu šířky textu na stránce.
  }
}

\noindent
\fbbox{
  \begin{minipage}{.45\textwidth}
    Toto je obsah prvního boxu o šířce 0,45-násobku šířky textu na stránce.
  \end{minipage}
}
\rule{1ex}{2pt}    % tento obdélník bude vidět
\fbbox{
  \begin{minipage}{.5\textwidth}
    Toto je obsah druhého boxu širokého polovinu šířky textu na stránce.
  \end{minipage}
}
```

Zde vidíme výsledek předchozího kódu:

Toto je text, který bude zarovnán do dvou sloupců, protože v parametru prostředí multicols je číslo 2. Kdyby tam bylo třeba číslo 4, tak by byl text zarovnaný do čtyř sloupců. V tomto prostředí mohou být trochu problémy třeba s popisky obrázků nebo poznámkami pod čarou.

Toto je obsah prvního boxu o šířce 0,45-násobku šířky textu na stránce.

Toto je obsah prvního boxu o šířce 0,45-násobku šířky textu na stránce.

Toto je obsah druhého boxu širokého polovinu šířky textu na stránce.
--

Toto je obsah druhého boxu širokého polovinu šířky textu na stránce.
--

2.8 Křížové odkazy

Křížové odkazy se dělají velice jednoduše a rychle. Zde si ukážeme, jak se vytvářejí odkazy na kapitolu nebo konkrétní místo v dokumentu (číslo strany), a také na číslované prostředí, které jsme vytvořili pomocí příkazu `\newtheorem`.

Odkaz na položku číslovaného seznamu jsme viděli v kapitole 2.4 na straně 55, odkazy na obrázek a tabulku probereme v kapitole 2.10.5.

Při práci s odkazy se může objevit jedno nebo i více varovných hlášení (LaTeX Warning: Reference ‘nejakyodkaz’ on page xx undefined ...). Toho si nemusíme všimnout, protože když si označíme místo, na které chceme odkazovat, je zaregistrováno až při překladu samotném, ale při adresování odkazu $\LaTeX$ vychází z informací získaných při předchozím překladu. Tedy při objevení varovných hlášení vztahujících se k odkazům prostě *znovu přeložíme*. Pokud se varovná hlášení objevují dále, je v názvu odkazu překlep.

Místo v dokumentu (záložku) pro jakýkoliv typ odkazu označíme příkazem

```
\label{nazev}
```

kde *nazev* je název označeného místa, který budeme v různých odkazech používat. Nepoužíváme v něm českou diakritiku (tedy píšeme „cesky“), ani mezery, tečky či většinu dalších oddělovačů. Naopak velmi často se v názvu objevuje dvojtečka jako užitečný oddělovač zkratky pro typ odkazu a zbytku názvu. Například

```
\label{kap:obsluhaprogramu},      \label{tab:obsluhaprogramu},
\label{obr:obsluhaprogramu},      \label{popis:obsluhaprogramu}
```

jsou různá místa v dokumentu (začátek kapitoly, obrázek, tabulka, popis), u kterých nám fantazie či spoléhání na vlastní paměť nestačily na různé názvy, ale pro přehlednost jsou odlišena jakousi předponou. Není to nutné, ale je to praktické a přehledné. Netřeba připomínat, že každý název musí být jedinečný – kdyby dvě místa byla pojmenovaná stejně, vznikl by jistý chaos při rozhodování, na které z nich se vlastně daným názvem odkázat.

Pokud chceme získat *číslo kapitoly*, ve které se nachází určité pojmenované místo, použijeme příkaz

```
\ref{nazev}
```

kde *nazev* je název pojmenovaného místa v příkazu `\label`. Pojmenované místo se může nacházet kdekoli v kapitole, jen ne uvnitř prostředí, kde jsou čítače, na které se dá také odkazovat (třeba číselný seznam nebo prostředí označovaná čísla, jako je třeba v tomto dokumentu prostředí pro příklady nebo výpisy kódu).

Číslo stránky, na které se pojmenované místo nachází, dostaneme příkazem

```
\pageref{nazev}
```

Význam je tentýž jako u příkazu `\ref`. Oba příkazy můžeme kombinovat, což se hodí zejména, pokud pojmenované místo a odkaz jsou na různých stránkách:

... je v kapitole `\ref{nazev}` na straně `\pageref{nazev}`

Odkazy na prostředí označené číslem se dělají úplně stejně – za začátek prostředí umístíme příkaz `\label` a pro odkaz použijeme příkaz `\ref`, případně `\pageref`.

Příklad 2.10 Pokud je v preambuli dokumentu definováno prostředí `příklad` jako na začátku výpisu 2.23, můžeme se na ně jednoduše odkazovat podle textu dále.

Výpis č. 2.23

```
\newtheorem{tpříklad}{Příklad}[chapter]
% (ve třídě article by místo chapter bylo section)
\newenvironment{příklad}%
  {\begin{tpříklad}\rule{1pt}{0pt}\normalfont}%
  {\end{tpříklad}}
...
\label{kap:vytvoreniiodkazu}
...
\begin{příklad}\label{pri:vytvoreniiodkazu}
text nějakého příkladu
\end{příklad}
...
V-příkladu \ref{pri:vytvoreniiodkazu} na straně
\pageref{pri:vytvoreniiodkazu} v-kapitole
\ref{kap:vytvoreniiodkazu} jsme mohli vidět, že ...
```

Příklad 2.10 text nějakého příkladu

...

V příkladu 2.10 na straně 70 v kapitole 2.8 jsme mohli vidět, že ...

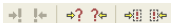
2.9 Jak na chyby

Snadnost nalezení chyb taky trochu závisí na editoru, který používáme. Po přeložení T_EX vytváří soubor s příponou LOG, ve kterém jsou uvedeny veškeré informace o překladu včetně určení chyb. Tento soubor většina editorů přímo určených přímo pro T_EX zobrazí v okně nebo v podokně, v případě programu T_EXnicCenter jde o oblast u spodního okraje okna. Dovíme se zde především, zda jsou v dokumentu chyby, varování a přesahy:

- *chyba* (Error) je takové selhání, které (většinou) znemožní vytvořit výsledný soubor, třeba překlep v příkazu, u začátečníků je častou chybou „připojení“ příkazu bez parametrů k následujícímu slovu (například `\larginějakéslovo`),
- *varování* (Warning) je „menší“ selhání, které obvykle nezabrání vytvoření výsledného souboru, ale „něco není v pořádku“; často se jedná o použití příkazu na nevhodném místě (třeba příkaz pro matematický režim se objeví mimo matematická prostředí) nebo chybějící řez pro používaný font, tyto problémy T_EX sám obvykle nějak vyřeší (provede náhradu nebo místo objektu vytvoří prázdné místo),
- *přesah* (Bad Box, vadný box) vzniká přesáhnutím objektu (včetně textu) tam, kde už nemá co dělat; jde většinou o slova, která se nedaří rozdělit na konci řádku tak, aby mezery mezi slovy zůstaly opticky vhodně malé či velké, ostatně přesáhnutí okraje stránky o milimetr většina lidí ani nepostřehne.

Chyby bychom měli rozhodně řešit, jen málokterá je zcela nedůležitá. Varování a přesahy je doporučeno řešit, ale není to už tak bezpodmínečně nutné. Některá varování a přesahy se obvykle neřeší nebo je řešení triviální:

- varování „There were undefined references.“ a „Label(s) may have changed ...“ se objeví vždy, když použijete příkaz pro pojmenování místa v dokumentu a případně toto pojmenování hned použijete pro křížový odkaz; stačí znovu provést překlad a tato varování zmizí (pokud nezmizí, máme buď v pojmenování nebo v křížovém odkazu překlep),
- varování „LaTeX Font Warning: Font shape ... not available ...“ se někdy objevuje, pokud nastavíme jiný než standardní font a chceme v něm používat řezy (některé fonty nejsou pro určité řezy definovány); tento problém $\LaTeX$ vyřeší přesně tak, jak píše dále v chybovém hlášení – na daném místě použije jiný (podobný) řez,
- hlášení o přesahu „Underfull \hbox ...“ znamená většinou problém s rozdělením slova; můžeme pomoci buď příkazem `\hyphenation` (viz nápověda), kde jako parametr použijeme problémová slova s naznačením možnosti rozdělení (na místech, kde lze slovo rozdělit, vložíme spojovník -, slova oddělujeme mezerou), nebo jednodušeji přímo ve slově, které vyvolává hlášení, na vhodném místě rozdělení vložíme dvojnásobek \- (například větší \-nu) – na konci řádku se slovo rozdělí, ale kdyby se náhodou posunulo dál od konce řádku, spojovník se nevypíše,
- při použití některých balíčků občas dostáváme varování a hlášení o vadných boxech, i když je víceméně všechno v pořádku; je to většinou způsobeno nekonzistentnostmi mezi balíčky, obvykle si těchto problémů nemusíme všimnout.

Zvláště pokud je problémů více, hodí se možnost pohybu mezi nimi. U programu $\TeX$ nicCenter najdeme tlačítka na horním panelu nástrojů napravo od „překládacích tlačítek“ – . Klepáním na tato tlačítka se pohybujeme ve zdrojovém souboru mezi chybami, varováními a přesahy a opravujeme. První dvě tlačítka nás v dokumentu přesouvají mezi chybami, další dvě mezi varováními a poslední dvě mezi přesahy.

Tímto způsobem se dostaneme vždy k *odstavci*, ve kterém je chyba, ne přímo ke konkrétnímu řádku. Najít konkrétní místo chyby nám pomůže výpis již zmíněného LOG souboru, ve kterém se přesouváme automaticky zároveň s přesouváním mezi chybami ve zdrojovém souboru.

2.10 Oživení textu

2.10.1 Vkládané obrázky

Pro vkládání již hotových obrázků jsou v $\LaTeX$ u dvě základní možnosti. První, použití balíčku `graphics`, je považována za jednodušší, ale jsme omezeni na vkládání obrázků s příponou PS a EPS při překladu přes formát PS a na obrázky s příponou PDF při překladu přímo do PDF. Druhou možností je balíček `graphicx`, kde můžeme při překladu přímo do PDF vkládat kromě jiného také obrázky s příponou JPG a PNG.

Nejdřív se podíváme na balíček `graphics`. Načítá se s nepovinným argumentem určujícím typ překladu. Pokud překládáme nejdřív do formátu PS a pak do PDF (v T_EXnicCenter LaTeX=>PS=>PDF), bude příkaz pro načtení vypadat takto:

```
\usepackage[dvips]{graphics}
```

Při překladu přímo do PDF (v T_EXnicCenter LaTeX=>PDF) použijeme jako parametr `pdftex`⁸.

Hotový obrázek vložíme do dokumentu příkazem

```
\includegraphics{název souboru}
\includegraphics*[x1,y1][x2,y2]{název souboru}
```

Druhý tvar příkazu umožní vložit do dokumentu pouze výřez obrázku (zadáme nejdřív souřadnice levého dolního rohu a pak pravého horního rohu v souřadnicích původního obrázku, počátek soustavy je vlevo dole).

Příkaz	Význam
<code>\rotatebox{úhel}{objekt}</code>	otočí objekt o zadaný úhel
<code>\scalebox{hnásobek}[vnásobek]{objekt}</code>	roztáhne objekt dle násobků
<code>\resizebox{šířka}{výška}{objekt}</code>	roztáhne objekt na zadané rozměry

Tabulka 2.6: Další příkazy balíčku `graphics`

První příkaz v tabulce 2.6 otočí objekt (cokoliv – text nebo třeba obrázek vložený pomocí `\includegraphics`) o zadaný úhel ve stupních. Druhý a třetí příkaz roztáhne nebo smrští objekt podle zadaných parametrů.

V příkazu `\scalebox` zadáváme násobnost, kdežto v příkazu `\resizebox` cílové rozměry. Parametr `vnásobek` je nepovinný, když ho neudáme, použije se i pro vertikální násobnost také parametr `hnásobek` (tedy poměry stran zůstanou zachovány).

U příkazu `\resizebox` může být jedním z parametrů `šířka` a `výška` symbol `!`. Tento parametr se dopočítá tak, aby si obrázek zachoval poměry stran.

Příklad 2.11 Na výpisu 2.24 vidíme způsob použití výše uvedených příkazů. Obrázek je nejdřív vložen bez jakýchkoliv úprav a potom pozměněn zadáním výřezu a příkazy `\rotatebox`, `\scalebox` a `\resizebox`. Potom je příkaz `\rotatebox` použit na běžný text. Abychom mohli pracovat s více řádky najednou, je text vložen v příkazu `\parbox`. Obrázky jsou odděleny příkazem `\hfill`, aby byly rovnoměrně rozloženy na celou šířku stránky.

Výpis č. 2.24

```
\includegraphics{obr/slunecnice.eps} \hfill
\includegraphics*[10,0][80,100]{obr/slunecnice.eps} \hfill
\rotatebox{15}{\includegraphics*[30,30][80,70]{obr/slunecnice.eps}} \hfill
\scalebox{.5}{\includegraphics{obr/slunecnice.eps}} \hfill
\resizebox{5em}{!}{\includegraphics{obr/slunecnice.eps}} \hfill
\rotatebox{90}{\parbox{8em}{\large\itshape Obrázek\slunecnice}}
```

⁸Parametr nemusíme uvádět, pokud ho napíšeme jako jeden z nepovinných parametrů příkazu `\documentclass`, například: `\documentclass[dvips,11pt,a4paper]{article}`.



V příkladu 2.11 jsme si mohli všimnout, že náš obrázek je uložen v adresáři (složce) obr, který se pak nachází v tom adresáři, kde je překládáný soubor s příponou TEX.

Jak bylo výše naznačeno, přípona souboru s obrázkem musí být PS nebo EPS při překladu přes formát PS, a nebo PDF při překladu přímo do formátu PDF. . Konverzi můžeme provést v běžných lepších grafických editorech, pokud je máme k dispozici (jsou obvykle dost drahé), nebo pomocí některého z volně dostupných programů *ImageMagick*⁹ a *Gimp*¹⁰.

Gimp má grafické prostředí, tedy rozhodně není problém konverzi provést, ale je hodně rozsáhlý (také místem, které zabere na disku) a je považován za snadno dostupný ekvivalent Adobe Photoshopu. ImageMagick je souhrn malých nenáročných programů, pro konverzi budeme používat program `convert`. Konverze do formátu EPS probíhá takto:

```
convert původní.soubor.přípona nový.soubor.eps
```

Obrázek, se kterým jsme pracovali na straně 73, měl původně příponu JPG. Proto bylo třeba ho převést do EPS takto:

```
convert slunecnice.jpg slunecnice.eps
```

Pokud místo balíčku `graphics` použijeme balíček `graphicx`, získáme některé další možnosti (například u otáčení lze kterýkoliv bod obrázku zvolit za střed otáčení), tvar parametrů je trochu jiný, navíc při překladu přímo do pdf máme větší svobodu volby přípony souboru. Balíček se načítá stejným způsobem (tj. obvykle zadáváme `dvips` nebo `pdftex`).

Nejdůležitějším příkazem balíčku je příkaz obdobný předchozímu, jen parametry jsou trochu jiné:

```
\includegraphics[parametry oddělené čárkou]{soubor.přípona}
\includegraphics*[parametry oddělené čárkou]{soubor.přípona}
```

⁹Program ImageMagick zvládá nejrůznější úpravy (nejen) obrázků. Získáme ho například na adrese <http://www.imagemagick.org/script/binary-releases.php>, vybereme variantu pro Linux nebo Windows, pro Windows raději statickou variantu. Na uvedených WWW stránkách je také návod k instalaci a použití.

¹⁰České stránky programu Gimp včetně nejrůznějších užitečných informací najdeme například na stránkách <http://www.gimp.cz/>, při stahování volíme raději stabilní verzi.

V seznamu parametrů oddělených čárkami se píše vždy název parametru následovaný rovnítkem a hodnotou. Jako parametry můžeme použít:

- výřez obrázku, který se má vložit, zadáváme postupně čtyři čísla – souřadnice levého dolního rohu a pak pravého horního rohu; formát je trochu jiný než u předchozího: `bb= x_levý y_dolní x_pravý y_horní`, pro správnou funkci tohoto parametru použijeme variantu příkazu s hvězdičkou,
- úhel otočení ve stupních: `angle=stupně`,
- šířka (obrázek se roztáhne nebo smrští), musíme zadat i jednotku (pt, cm, atd.): `width=šířka`,
- výška (podobně jako šířka): `height=výška`,
- zvětšení nebo zmenšení obrázku: `scale=poměr` (výchozí hodnota je 1, tedy zadáváme násobnost původní velikosti),
- atd. (viz dokumentace balíčku `graphicx`).

Například:

```
\includegraphics*[bb=10 0 80 100,scale=0.75]{slunecnice.eps}
```

V balíčku najdeme i další příkazy – `\rotatebox`, `\scalebox`, atd. Tyto příkazy mají podobný význam jako stejně pojmenované příkazy v balíčku `graphics`, ale parametry jsou trochu jiné a také máme k dispozici více možností nastavování vlastností zpracovávaných objektů.

2.10.2 Nákresy

Pro vytváření jednoduchých grafických objektů používáme prostředí `picture`. Používá se s udáním rozměrů „plátna“, které (trochu nezvykle) píšeme do *kulatých závorek* jako parametr prostředí, a to bez udání jednotky (standardně je jednotkou typografický bod).

```
\begin{picture}(šířka,výška)
...
\end{picture}
```

Dovnitř prostředí píšeme příkazy vykreslující jednotlivé objekty (rámečky, čáry–úsečky, vektory, kružnice, ovály, atd.). Každý objekt musíme umístit na určitou pozici příkazem

```
\put(x,y){objekt}
```

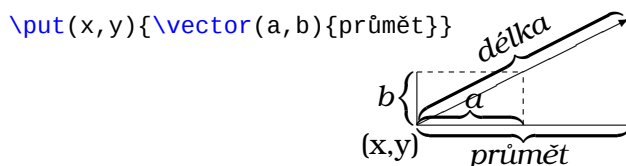
kde x a y jsou souřadnice, na které chceme objekt umístit. Bod $(0, 0)$ je v levém dolním rohu.

Dvojčíslí v kulatých závorkách se používá také v některých příkazech pro kreslení objektů. Někde znamená rozměry (šířku a výšku) objektu, jako třeba u rámečků (příkazy `\framebox`, `\dashbox` a `\makebox`), u oválů (příkaz `\oval`) délky vodorovně

a svislé osy, jindy zase určení úhlu sklonu, a to pro čáry a vektory (příkazy `\line` a `\vector`).

Čísla pro úhel sklonu a parametr pro výpočet délky čáry a vektoru se určují podle obrázku 2.7. Čísla a a b musí být celá čísla z intervalu $< -6,6 >$ u čar a z intervalu $< -4,4 >$ u vektorů. Různými kombinacemi těchto čísel dosáhneme různých sklonů.

Můžeme si všimnout, že nezadáváme přímo délku (kromě svislé čáry či vektoru, tam ano), ale *délku průmětu* na osu x .



Obrázek 2.7: Určení sklonu čáry a vektoru

Příklad 2.12 Ukážeme si použití prostředí `picture` a vkládání různých typů objektů. Vysvětlení parametrů příkazů najdeme v tabulkách 2.7, 2.8 a 2.9 od strany 76. V příkazech si všimněte, kdy se u čísla udává jednotka.

Výpis č. 2.25

```
\begin{picture}(160,150)
\put(0,0){\framebox(160,150){}}
\put(5,5){\dashbox{3}(150,140){}}
\put(10,10){\small Jeden řádek.}
\put(10,20){\makebox(60,10){\small Jeden řádek.}}
\put(75,10){\framebox(68,28)%
{\parbox{58pt}{\small Případně více řádků.}}}

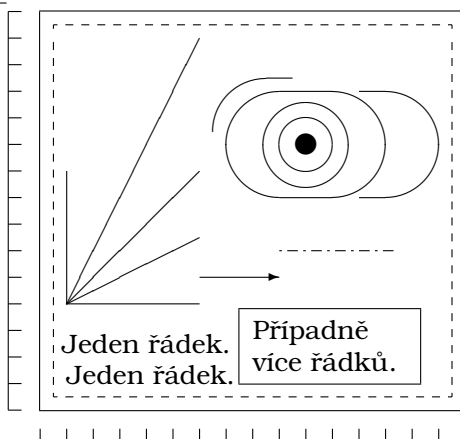
\put(10,40){\line(0,1){50}}
\put(10,40){\line(1,0){50}}
\put(10,40){\line(1,1){50}}
\put(10,40){\line(1,2){50}}
\put(10,40){\line(2,1){50}}

\put(60,50){\vector(1,0){30}}

\put(100,100){\circle{20}}
\put(100,100){\circle{30}}
\put(100,100){\circle*{8}}

\put(100,100){\oval(60,40)}
\put(120,100){\oval(60,40)[r]}
\put(95,105){\oval(60,40)[lt]}

\multiput(90,60)(9,0){5}{\line(1,0){1}}
\multiput(93,60)(9,0){5}{\line(1,0){4}}
\end{picture}
```



Pro zpřehlednění obrázku jsou připojena „pravítka“ s díly po deseti bodech. Je to zároveň ukázka toho, že můžeme bez problémů kreslit i mimo oblast vyhrazenou v parametru prostředí `picture`. Pravítka jsou vykreslena těmito příkazy:

Výpis č. 2.26

```
\multiput(0,-12)(10,0){17}{\line(0,1){5}}      % vodorovné pravítko
\put(0,-12)\line(1,0){160}}
\multiput(-12,0)(0,10){16}{\line(1,0){5}}      % svislé pravítko
\put(-12,0)\line(0,1){150}}
```

To, jaké rozměry pro prostředí `picture` zadáme, určuje, kolik místa bude pro ně na stránce rezervováno. Protože není problém kreslit i mimo prostředí (to můžeme vidět na výpisu 2.26, kde jsou použity záporné souřadnice), záleží na nás, kolik místa chceme mít vynecháno. Dokonce není problém vytvořit prostředí `picture` s nulovými rozměry (v parametru `(0,0)`). Toho se využívá například při umístění obrázku do záhlaví nebo vytvoření „vodoznaku“ na pozadí stránky.

Příkaz	Význam
<code>\put(x,y){objekt}</code>	vloží objekt na dané souřadnice
<code>\multiput(x,y)(dX,dY){počet}{objekt}</code>	vloží objekt počet-krát, začíná na souřadnicích (x, y) a v každém kroku se posune o dX na ose X a o dY na ose Y
<code>\qbezier[počet](x1,y1)(x2,y2)(x3,y3)</code>	vykreslí Bezierovu křivku (začíná v prvním bodě, končí v třetím, druhý určuje zakřivení), v nepovinném parametru můžeme určit, že se nevykreslí celá křivka, ale pouze daný počet bodů z ní

Tabulka 2.7: Příkazy pro kreslení, které se píšou přímo do prostředí `picture`

Příkaz	Význam
<code>\line(poměrX,poměrY){průmět}</code>	čára (úsečka) o zadaném poměru stran a délce průmětu na osu X (u svislé čáry její délce), čísla pro poměr jsou celá čísla z intervalu $< -6, 6 >$
<code>\vector(poměrX,poměrY){průmět}</code>	vektor, jako u čáry, jen interval je $< -4, 4 >$
<code>\circle{průměr}</code>	kružnice o daném průměru
<code>\circle*{průměr}</code>	vyplněný kruh o daném průměru
<code>\oval(osaX,osaY)[část]</code>	ová se zadanými délkami vodorovné a svislé osy, v nepovinném parametru můžeme určit část, která se má vykreslit (jedno nebo dvě písmena z t, b, l, r)
<code>\shortstack[poz]{řádky}</code>	řádky textu, poz je zarovnání vnitřku (l, r, c)

Tabulka 2.8: Příkazy objektů vkládaných do parametrů příkazů `\put` a `\multiput`

V tabulce 2.8 máme seznam některých příkazů, které vytvářejí různé objekty. Není to úplný seznam, protože zde můžeme používat prakticky (téměř) cokoli, co jsme až dosud z $\text{\LaTeX}$ probrali včetně formátovaného textu, vkládaných obrázků i různě upravovaných (oříznutí, rotace, apod.), dokonce i vnořené prostředí `picture`. To je výhoda zejména tehdy, když máme skupinu objektů, jejichž umístění je vzájemně závislé a zatím není definitivní. Když je umístíme do vnořené prostředí `picture`, přesouváme celé toto prostředí bez „rozházení“ objektů v něm umístěných.

Příkaz	Význam
<code>\framebox(š,v)[poz]{obsah}</code>	rámeček s plným orámováním o daných rozměrech (šířka, výška), do obsahu lze dát cokoli (i text nebo čáru), nepovinný parametr určuje zarovnání obsahu uvnitř (jedno nebo dvě písmena z t, b, l, r, c)
<code>\dashbox{element}(š,v)[poz]{obsah}</code>	rámeček s čárkováním orámováním, platí pro něj totéž co pro <code>\framebox</code> , jen navíc v parametru <code>element</code> určujeme délku čárek
<code>\makebox(š,v)[poz]{obsah}</code>	rámeček bez orámování, používá se pro přesnější umístění například textu

Tabulka 2.9: Příkazy pro rámečky v prostředí `picture`

Pro víceřádkový text se používá buď kombinace příkazů `\makebox` a `\parbox` (ten známe z kapitoly 2.7) nebo příkaz `\shortstack`. Na výpisu 2.27 vidíme drobné rozdíly v používání těchto možností. První možnost dovoluje snadnější manipulaci s textem samotným včetně výchozího zarovnání do bloku, řádky jsou zalamovány automaticky. Při použití druhé možnosti určujeme zarovnání řádků v nepovinném parametru, zalamování musíme provést ručně (ale zato nemusíme určovat šířku číselně) a po každém zalomení řádku je třeba znovu uvést formátovací příkazy.

Výpis č. 2.27

```
\begin{picture}(98,80)
\put(0,0){\framebox(98,80){}}
\put(1,45){\makebox(95,30){\parbox{90pt}{\small Toto je
nějaký text zarovnaný \itshape do více řádků.}}}
\put(3,3){\shortstack[l]{\small Toto je nějaký text\\
\small zarovnaný \itshape do více\\ \small\itshape řádků.}}
\end{picture}
```

Toto je nějaký text
zarovnaný *do více*
řádů.

Toto je nějaký text
zarovnaný *do více*
řádů.

Zde probírané možnosti prostředí `picture` nejsou plným popisem, podrobnější informace jsou například v nápovědě, [T31], [T43] (krásné příklady, v odkazech nahoře) a [T39]. Existují také balíčky, které rozšiřují možnosti prostředí `picture` – například `epic`, `eepic`, `pictex`, `xypic`, `pmgraph`, sada balíčků `pstricks` (zajímavé jsou tutoriály na [T44]).

2.10.3 Tabulátory

Pro tabulátory máme v L^AT_EXu prostředí `tabbing`. Uvnitř prostředí používáme především příkazy na nastavování tabulátorových zarážek a přecházení mezi nimi a samozřejmě příkazy pro přechod na nový řádek. Máme možnost také uložit pozice tabulátorů a kdykoliv je znovu využít, ale pouze v tomtéž prostředí `tabbing`. Toho se využívá především při změně nastavení zarážek a opětovném návratu k předchozímu nastavení.

Příkaz	Význam
<code>\=</code>	nastavení tabulátorové zarážky na této pozici
<code>\></code>	přechod na další zarážku (jako bychom stiskli klávesu <code>[Tab]</code>)
<code>\'</code>	předchozí text je svým pravým okrajem (s určitou malou mezerou) zarovnán k předchozí zarážce
<code>\'</code>	všechn další text až do symbolu konce řádku posune k pravému okraji, na řádku nesmí následovat žádné další tabulátorové příkazy
<code>\\</code>	přechod na nový řádek, v prostředí <code>tabbing</code> zůstávají tabulátorové zarážky nastaveny
<code>\kill</code>	pokud je tento příkaz použit na konci řádku místo <code>\\</code> , tento řádek se nevysází, ale všechny na něm nastavené tabulátorové zarážky zůstávají nastaveny
<code>\pushtabs</code>	uloží pozice všech zarážek do zásobníku tabulátorových zarážek
<code>\poptabs</code>	vyjme poslední uložené pozice zarážek ze zásobníku tabulátorových zarážek a nastaví je

Tabulka 2.10: Příkazy pro tabulátory

Příkazy `\pushtabs` a `\poptabs` jsou párové (ke každému `\pushtabs` musí být jeden `\poptabs`) a můžeme je vnořovat podobně jako symboly hranice skupiny `{ }`.

Obvyklý způsob použití je nastavení zarážek pomocí `\=` na zvláštním řádku, kde pro jednotlivé části použijeme vždy z daného „sloupce“ nejširší část přes všechny řádky (případně ještě rozšíříme) a na konci tohoto zvláštního řádku napíšeme příkaz `\kill`. Tento řádek bude sloužit pouze pro nastavení, nevysází se. Další řádky již píšeme běžným způsobem, pro pohyb k dalším zarážkám používáme příkazy `\>`, `\'` a `\'` a řádky končíme příkazem `\\`.

Příklad 2.13 Zarovnání k tabulátorovým zarážkám si ukážeme na fragmentu úlohy 5 ze strany 4. Text je zarovnán k zarážce vlevo, čísla vpravo.

Výpis č. 2.28

```
\begin{tabbing}
Počet xx \= zajíců xx 126 \= x \kill
Počet \> sobů \> 126 \'\
\> lišek \> 12 \'\
\> zajíců \> 54 \'\[.8ex]
\textbf{Celkem zvířat}\>\> \textbf{192}\
\end{tabbing}
```

Počet	sobů	126
	lišek	12
	zajíců	54
Celkem zvířat		192

Prostředí `tabbing` nelze vnořovat, kromě případu, kdy je „vnořené“ prostředí uzavřeno do příkazu `\parbox`.

Stejného vizuálního efektu, jako pomocí prostředí `tabbing`, dosáhneme například použitím tabulek (viz kapitola 2.10.4) nebo skládáním prostředí `minipage` či příkazů `\parbox`. Zde pak máme například možnost používat pružné čáry plné či tečkované, u tabulek také podtržení a shrnutí více buněk. Ve výpisu 2.29 vidíme kombinaci dvou použití příkazu `\parbox` a pružné čáry.

Výpis č. 2.29

```
\parbox{3em}{ Počet}
\parbox[t]{8em}{%
sobů\dotfill 126\\
lišek\dotfill 12\\
zajíců\dotfill 54 }
```

Počet	sobů.....	126
	lišek.....	12
	zajíců.....	54

2.10.4 Tabulky

Pro sazbu tabulek používáme prostředí `tabular` (pozor, neplést si s prostředím pro tabulátory) s následujícím způsobem použití:

```
\begin{tabular}{určení sloupců} ... \end{tabular}
```

V parametru prostředí musíme určit počet sloupců s jejich zarovnáním a oddělovače.

Zarovnání sloupců se zadává písmeny `l`, `r`, `c`, `p` (jeden řádek zarovnaný doleva, doprava, vycentrováný, a nebo odstavec zarovnaný do bloku), při použití `p` určujeme šířku odstavce v parametru, například `p{2cm}` je sloupec široký 2 centimetry, ve kterém jsou odstavce zarovnané do bloku.

Oddělovač může být buď čára (zadáme symbol `|`) nebo jakýkoliv jiný symbol včetně matematických. Dvojitou čáru zadáme dvěma symboly pro čáru za sebou, tedy `||`, jiný oddělovač než čáru určíme příkazem `@{znaky}`. Když oddělovač nezadáme vůbec, sází se pro oddělení sloupců mezera, pokud nechceme ani tu mezeru, zadáme jako oddělovač `@{}`.

Protože oddělovače oddělují sloupce a sloupce jsou určeny svým zarovnáním, parametr `určení sloupců` se zadává například takto:

```
{|l|c|p{.3\textwidth}|r|}
```

první sloupec je zarovnán vlevo, druhý na střed, třetí je odstavec o šířce přibližně třetiny stránky a čtvrtý je zarovnán doprava, mezi nimi jsou jako oddělovače použity čáry, stejně jako na stranách celé tabulky,

```
{|l||r@{${}$}r@{${}$}r@{${}$}r|}
```

první sloupec je zarovnán doleva, pak následuje dvojitá čára, zbylé sloupce jsou zarovnány doprava (zřejmě pro čísla) a mezi nimi jsou použity jako oddělovače symboly `+` a `=`,

`{l|r@{,}l}` první sloupec je zřejmě použit pro běžný text, další dva dohromady tvoří reálné číslo s desetinnou čárkou (druhý sloupec zaznamenává celou část, třetí desetinnou část).

Vodorovnou čáru tvoříme příkazem `\hline`, dvojitou čáru zdvojením tohoto příkazu. Vodorovnou čáru přes daný počet sloupců získáme příkazem `\cline{od-do}`, kde zadáme čísla sloupců, mezi kterými má čára vést.

Více sloupců na jednom řádku (buněk) sloučíme příkazem

`\multicolumn{počet sloupců}{zarovnání}{obsah buňky}`

Uvnitř tabulky používáme k oddělení sloupců příkaz `\\` případně následovaný příkazy `\hline` či `\cline`, buňky na řádku oddělujeme symbolem `&`.

Příklad 2.14 Použití základních možností tabulky si ukážeme na jednoduchých příkladech vycházejících z výpisu 2.28 na straně 78. Ve výpisu 2.32 je v zarovnání na okrajích příkaz `@{}`, aby podtrhávací čára „nepřečnívala“. Bez použití tohoto příkazu by se místo okrajové čáry vložila mezera, jenž my nechceme ani tu mezeru.

Výpis č. 2.30

```
\begin{tabular}{|l|r|}\hline
Počet & sobů & 126\\
& lišek & 12 \\
& zajíců & 54 \\ \hline\hline
\multicolumn{2}{|l|}{\bfseries Celkem zvířat}&
\bfseries 192\\ \hline
\end{tabular}
```

Počet	sobů	126
	lišek	12
	zajíců	54
Celkem zvířat		192

Výpis č. 2.31

```
\begin{tabular}{|l|r|}\hline
\bfseries Zvíře & \bfseries Počet\\ \hline\hline
sob & 126\\ \hline
liška & 12 \\ \hline
zajíc & 54 \\ \hline\hline
\bfseries Celkem zvířat & \bfseries 192\\ \hline
\end{tabular}
```

Zvíře	Počet
sob	126
liška	12
zajíc	54
Celkem	192

Výpis č. 2.32

```
\begin{tabular}{@{}lp{7.7em}@{}}
Počet & sobů & \dotfill 126\\
& lišek & \dotfill 12 \\
& zajíců & \dotfill 54 \\ \hline
\multicolumn{2}{@{}l@{}}{\bfseries Celkem zvířat}
& \dotfill 192}
\end{tabular}
```

Počet	sobů		126
	lišek		12
	zajíců		54
Celkem zvířat			192

Příklad 2.15 V tomto příkladu se podíváme na zarovnání reálných čísel podle desetinné čárky. Celou část zarovnáme doprava, desetinnou doleva a před sečtením podtrhneme.

Výpis č. 2.33

```
\begin{tabular}{lr@{,}l}
Výpočet: & 1\,284&52\\
&716&4\\
&0&977\\
&35&0\\ \cline{2-3}
&2\,036&897
\end{tabular}
```

Výpočet: 1 284,52
716,4
0,977
35,0

2 036,897

Příklad 2.16 Ukážeme si vytvoření vlastních oddělovačů na jednoduchém „sčítacím“ příkladu. Ve výpisu 2.34 je kód příkladu, v tabulce 2.11 pak výsledek.

Výpis č. 2.34

```
\begin{tabular}{l||c@{~+~}c@{~+~}c@{~+=~}c}
\bfseries Jméno & Příklad 1 & Příklad 2 & Příklad 3 & Celkem\\ \hline\hline
\itshape Maximum: & \itshape 6&\itshape 7&\itshape 7&\itshape 20\\ \hline\hline
Blažková & 3 & 7 & 4 & 14\\ \hline
Hrubý & 2&3&1&6\\ \hline
Novák & 4 & 5 & 7 & 16\\ \hline
\dotso
\end{tabular}
```

Jméno	Příklad 1 + Příklad 2 + Příklad 3 = Celkem				
<i>Maximum:</i>	6	+	7	+	7 = 20
Blažková	3	+	7	+	4 = 14
Hrubý	2	+	3	+	1 = 6
Novák	4	+	5	+	7 = 16
...					

Tabulka 2.11: Ukázka tabulky s vlastními oddělovači

Ukázali jsme si pouze základní možnosti tvorby tabulek. Možností je více a navíc je můžeme rozšířit použitím různých balíčků, například `tabularx`, `supertabular`, `slashbox`, apod. Obvykle jde především o možnost vytvářet tabulky zasahující na více stran (zde ukázané tabulky mohou být pouze na jedné straně). Podrobnosti viz například [T31] nebo na prakticky každém odkazu ze seznamu literatury tohoto dokumentu vedoucím na některou učebnici $\text{\LaTeX}$ u.

2.10.5 Plovoucí prostředí

V tomto dokumentu jsme si mohli všimnout způsobu vkládání obrázků a tabulek. Tyto objekty jsou obvykle pojmenované (mají popisek s číslem), lze se na ně odkazovat a jsou umístěny tak, že sice někdy nebývají přesně tam, kde se o nich píše, ale v případě, že by přes ně mohl vést stránkový zlom (konec stránky), přesunou se na další stránku bez toho, aby místo nich na předchozí stránce zůstalo kus místa nevyužitého.

Tuto funkci v $\TeX$ u plní *plovoucí prostředí*. Nazývá se tak, protože „plave“ na stranách tak, aby se vyhnulo stránkovému zlomu a umístilo se tak, jak si autor přeje (pokud je to možné). Rozlišujeme dvě základní plovoucí prostředí – pro tabulky je to `table` a pro obrázky a jiné grafické objekty je to `figure`. Používají se stejně:

<code>\begin{table}[umístění]</code>	<code>\begin{figure}[umístění]</code>
<code>\centering</code> % vycentrování	<code>\centering</code> % vycentrování
... tady bude tabulka, třeba prostředí <code>tabular</code>	... tady bude objekt, třeba prostředí <code>picture</code>
<code>\caption{popisek tabulky}</code>	<code>\caption{popisek objektu}</code>
<code>\label{pojmenování pro odkazy}</code>	<code>\label{pojmenování pro odkazy}</code>
<code>\end{table}</code>	<code>\end{figure}</code>

V parametru umístění určujeme písmeny `h`, `t`, `b`, `p`, kde se má na stránce objekt umístit, pokud to půjde (na místě, na horním okraji stránky, dolním okraji a nebo na zvláštní stránce pouze pro plovoucí prostředí, obvykle na konci kapitoly nebo dokumentu). Píšeme více možností, $\TeX$ je zkouší zleva – když se nepovede umístit prostředí na místě, posune ho na horní okraj následující stránky apod., například `[htbp]`.

Tabulka 2.11 na straně 81 byla umístěna tímto kódem:

Výpis č. 2.35

```
\begin{table}[htb]
\centering
\begin{tabular}{...}
...
\end{tabular}
\caption{Ukázka tabulky s~vlastními oddělovači}
\label{tab:scitani}
\end{table}
```

Na tuto tabulku se můžeme odkázat takto: v tabulce `\ref{tab:scitani}` na straně `\pageref{tab:scitani}` ... – v tabulce 2.11 na straně 81 ...

Příklad 2.17 Umístíme obrázek slunečnice s popisem. Použijeme přitom obrázek, který jsme používali v kapitole 2.11 na straně 73.



Obrázek 2.8: Slunečnice

Kdyby tento obrázek byl uprostřed strany, byl by vložen a umístěn tímto kódem:

Výpis č. 2.36

```
\begin{figure}[htb]
\centering
\includegraphics{obr/slunecnice.eps}
\caption{Slunečnice}
\label{obr:slunecnice}
\end{figure}
```

Uvnitř plovoucího prostředí nemusí být jen jedna tabulka či obrázek. Můžeme tam vložit prakticky cokoliv (i text) a v jakémkoliv počtu, pak se objekty naskládají vedle sebe. Například v kapitole o MS Wordu je často používáno prostředí `figure`, ve kterém je kromě obrázku vloženého příkazem `\includegraphics` také prostředí `picture` o velmi malých rozměrech pro popisování jednotlivých částí vloženého obrázku.

Stejně jako u jiných témat, také u plovoucích prostředí najdeme mnoho rozšíření v podobě balíčků. Například balíčky `wrapfig`, `picins` a `floatflt` nám umožňují vkládat plovoucí prostředí do vícesloupcového textu, prostředí `minipage` či příkazu `\parbox`, nebo je obtékat.

Balíček `picins` má bohužel dokumentaci jen v němčině, ale některé jeho možnosti při vkládání objektů jsou pěkně popsány v [T31], v nápovědě nebo na internetu (stačí na <http://google.com> zadat heslo „picins documentation“ nebo jen „picins“). Zajímavý je především příkaz `\parpic`, který kromě zajištění obtékání objektu do něj vloženého umí například také kolem objektu vytvořit rámečky různých tvarů včetně oválného.

Balíček `floatflt` obsahuje prostředí `floatingfigure` a `floatingtable`, balíček `wrapfig` obsahuje prostředí `wrapfigure` a `wraptable`. Jsou podobné, ale o něco více možností nabízí balíček `wrapfig`. Obrázek 2.8 byl ve skutečnosti vložen tímto kódem (je načten balíček `wrapfig`):

Výpis č. 2.37

```
\begin{wrapfigure}[9]{l}{115pt} % přes 9 řádků, vlevo, šířka 115 bodů
\centering
\includegraphics{obr/slunecnice.eps}
\caption{Slunečnice}
\label{obr:slunecnice}
\end{wrapfigure}
Kdyby tento obrázek byl uprostřed strany ...
```

Balíčky `subfig` a `subfigure` dovoluují členit plovoucí prostředí na menší pojmenované části, tedy máme v jednom prostředí více obrázků a každý má vlastní popis. Pomocí balíčku `subfloat` vytváříme víceúrovňové číslování různých plovoucích prostředí, `captcont` rozděluje dlouhá plovoucí prostředí na více stránek a `ccaption` dovoluje kromě jiného vytvářet i nečíslované nebo vícejazyčné popisky.

2.10.6 Boxy

Box (rámeček) je oblast stránky, se kterou lze zacházet do určité míry samostatně. Existuje více různých druhů boxů a každý má své specifické vlastnosti. S některými jsme se už setkali, kromě příkazu `\fbox` a `\parbox` především v prostředí `picture` v kapitole 2.10.2 (jsou v tabulce 2.9 na straně 77)¹¹. Stejně pojmenované boxy, ale s trochu jinou syntaxí (způsobem použití) používáme i mimo nákresy v textu, a také některé další.

Z boxů již známe ten, do kterého dáváme objekty (včetně textu), které chceme srovnat na určitou šířku, objekty mohou být i na více řádcích:

```
\parbox[zarovnání od předchozího]{šířka}{obsah}
```

Další boxy používáme, když potřebujeme text či jiné objekty vhodně zarovnat či umístit, případně orámovat

```
\makebox[šířka][zarovnání]{obsah}
\mbox{obsah}
\framebox[šířka][zarovnání]{obsah}
\fbox{obsah}
```

Příkaz `\makebox` prostě zarovná (c, l, r) text v dané šířce (třeba vycentruje na tuto šířku). Zjednodušenou variantou je příkaz `\mbox`, kde nepotřebujeme žádné nepovinné parametry, používáme ho třeba tehdy, když některé slovo nebo několik slov určitě nechceme rozdělit na konci řádku, chceme je udržet celé „v kuse“. Další dva příkazy mají podobný význam, jen navíc přidávají rámeček kolem svého obsahu.

Nevýhodou těchto příkazů je, že nedokážou pracovat s více řádky. To se řeší vnořením obsahu do příkazu `\parbox` nebo do prostředí `minipage`, například takto:

```
\fbox{\parbox[zarovnání od předchozího]{šířka}{obsah}}
```

Další typ boxu slouží k vertikálnímu posouvání objektů. Někdy potřebujeme část textu, obrázek či cokoliv jiného posunout nahoru nebo dolů, pak použijeme tento příkaz:

```
\raisebox{zvýšení}{obsah}
```

Parametr zvýšení je délka, o kolik se má obsah posunout nahoru, když je délka záporná, posouvá se dolů. Opět při práci s více řádky musíme použít kombinaci s příkazem `\parbox` nebo prostředím `minipage`.

V boxech můžeme jako šířku či jinou míru použít také některé délkové registry. Často se používají například `\width` (šířka obsahu), `\height` (výška obsahu od základní čáry řádku nahoru), `\depth` (hloubka obsahu od základní čáry řádku dolů), `\totalheight` (celková výška obsahu – součet předchozích dvou).

¹¹Pozor na rozdíl mezi stejně pojmenovanými boxy pro text a pro prostředí `picture` – mají jiné parametry!

Příklad 2.18 Vyzkoušíme si použití příkazu `\framebox` s nepovinným parametrem, jeho kombinaci s příkazem `\makebox` a také zvyšování a snižování textu. Všimněte si u druhého boxu zadání dvojnásobku šířky obsahu. Pro orámování výsledku kódu výpisu je použita kombinace příkazu `\fbox` a prostředí `minipage`.

Výpis č. 2.38

Píšeme orámovaný text na
`\framebox[10em]{danou šířku}`
 10 čtverčků, a další v boxu širokém dvojnásobek
 tohoto `\framebox[2\width]{textu}`.

Máme orámované písmenko `\fbox{j}` a pak písmenko
`\fbox{\makebox[\totalheight]{j}}`
 ve čtvercovém rámečku.

% zvyšování a snižování objektů:

Píšeme `\raisebox{3pt}{zvýšený,}` normální a také
`\raisebox{-3pt}{snížený text.}`

Píšeme orámovaný text
 na danou šířku 10
 čtverčků, a další v boxu
 širokém dvojnásobek to-
 hoto textu.

Máme orámované písmen-
 ko j a pak písmenko j
 ve čtvercovém rámečku.

Píšeme zvýšený, normální
 a také snížený text.

Boxy lze také ukládat a znovu používat, to se týká i boxů v prostředí `picture`. Podrobnosti najdeme například v nápovědě nebo v [T31].

2.10.7 Barvy

Podporu barev přidáme balíčkem `color` s nepovinným parametrem `dvips` nebo `pdftex` podle toho, jestli chceme překládat přes formát DVI a PS (`LaTeX=>DVI` nebo `LaTeX=>PS=>PDF`) nebo přímo do PDF (`LaTeX=>PDF`). Máme možnost používat předdefinované barvy (s anglickými názvy) pro text, pozadí textu, boxy i kreslení, a také vytvářet si vlastní barvy.

Následující příkazy slouží k nastavení určité barvy pro úsek textu, a to pro samotný text, pozadí textu nebo obojí najednou:

```
\textcolor{barva}{text}
\colorbox{barva pozadí}{text}
\fcolorbox{barva rámečku}{barva pozadí}{text}
```

Existují i příkazy platné od místa jejich uvedení až do konce skupiny, slouží pro nastavení barvy textu (nebo barvy pro kreslení v prostředí `picture`) nebo pro nastavení barvy pozadí celé stránky (změna barvy pozadí stránky se projeví až po překládu do formátu PS nebo PDF):

```
\color{barva}
\pagecolor{barva pozadí}
```

Novou barvu vytvoříme následujícím příkazem, ve kterém zadáme název nové barvy, číselnou specifikaci a pokud možno také typ (obvykle se použije zkratka `rgb` pro barevnou kombinaci červené, zelené a modré nebo `gray` pro odstín šedě):

```
\definecolor{název barvy}{typ barvy}{specifikace}
```

Příklad 2.19 Ukážeme si použití příkazů pro barvení textu, jeho pozadí a kreslení, a také vytvoření vlastní barvy.

Výpis č. 2.39

```
\textcolor{green}{Zelený text.}

\fcolorbox{blue}{yellow}{\parbox{6em}{%
  \textcolor{red}{Červený text} %
  \color{blue} v~rámečku}}

\def\ctverecek{\rule{1em}{1em}}
\textcolor{black}{\ctverecek}
\textcolor{blue}{\ctverecek}
\textcolor{red}{\ctverecek}
\textcolor{green}{\ctverecek}
\textcolor{cyan}{\ctverecek}
\textcolor{yellow}{\ctverecek}

% barvy se definují čísla z intervalu <0,1>,
% vyšší číslo znamená světlejší barvu:
\definecolor{svmodra}{rgb}{.65,.65,1}
\definecolor{seda}{gray}{.4}
\colorbox{svmodra}{Světlemodré \color{seda} pozadí.}

\begin{picture}(1,100)
\color{red} % odteď se kreslí červenou
\qbezier(40,30)(50,0)(75,90)
\end{picture}
```



2.10.8 Hypertextové odkazy

Hypertextové odkazy vytváříme pomocí balíčku `url` nebo `hyperref`.

Použití balíčku `url` je jednoduché. Načteme ho příkazem `\usepackage` (není třeba zadávat žádné další parametry) a hypertextový odkaz vložíme příkazem

```
\url{adresa}
```

Zadaná adresa je vypsána neproporcionálním písmem („psací stroj“) a jsou zobrazeny znaky, které by jinak bylo nutné zobrazovat pomocí příkazů (například vlnovka nebo zpětné lomítko). Pokud si přejeme ještě dále adresu formátovat, musíme to udělat ručně. Například modrou barvu a podtržení nastavíme takto:

```
{\color{blue}\underline{\url{http://nejaka.adresa.cz}}}
```

Vypadá to jako hypertextový odkaz, ale nechová se to tak. Pro dokument, který se má tisknout, je to dostačující, ale pokud chceme, aby byl použitelný i v elektronické podobě včetně možnosti používat odkazy, musíme volit jiné řešení. Existuje program `dvipdfm`, který dokáže vyrobit dokument s těmito vlastnostmi, ale je nutno znát parametry, se kterými se má spustit.

Jednodušším řešením je použití balíčku `hyperref`, hypertextové odkazy fungují až v souboru PDF. Je důkladně popsán v dokumentaci, kterou snadno najdeme (viz kapitola 2.6.2). Je spíše určen pro překlad přímo do PDF (LaTeX=>PDF), ale lze ho použít i při překladu přes DVI a PS (LaTeX=>PS=>PDF) s tím, že některé volby budou dělat trochu problémy.

Balíček `hyperref` nabízí mnoho voleb (některé se vlastně ani netýkají hypertextových odkazů), i proto stojí za to prostudovat si dokumentaci. Volby se zadávají buď při načítání balíčku příkazem `\usepackage`, nebo v příkazu `\hypersetup`.

Při překladu přes PS můžeme balíček načíst například s těmito volbami:

```
\usepackage[dvips,colorlinks,linkcolor={blue},%
citecolor={green},urlcolor={blue}]{hyperref}
```

Volba `dvips` je nutná při překladu přes DVI a PS, volba `colorlinks` znamená, že se odkazy mají zobrazovat barevně (byly by však aktivní i bez barevného zvýraznění), další volby určují barvy pro jednotlivé druhy odkazů. V našem případě jsou odkazy vedoucí dovnitř dokumentu (třeba na určitou stranu, kapitolu nebo v obsahu) modré, odkazy do seznamu literatury zelené a odkazy „ven“ (internetové) modré.

Názvy barev (předdefinovaných) můžeme používat i bez načtení balíčku `color`, ale pokud chceme vytvářet vlastní barvy, musíme tento balíček do dokumentu načíst (a barvu samozřejmě definovat ještě před jejím případným použitím ve volbách balíčku `hyperref`).

Balíček `hyperref` také umožňuje vytvářet záložky podle kapitol dokumentu, tyto záložky se pak zobrazí v některých prohlížečích (například Adobe Reader) vlevo vedle dokumentu na záložce Záložky. Při překladu přímo do PDF nebývají problémy, ale pokud překládáme přes PS, musíme zajistit, aby se správně zobrazovaly české znaky (tím zároveň vyřešíme i problémy s češtinou při používání aktivních odkazů). Použijeme balíček `inputenc` s vhodnou volbou (například `latin2`) a u balíčku `hyperref` volbu `unicode`.

Celkově po vyřešení vlastních barev, češtiny a záložek (včetně čísel kapitol) může načtení balíčku vypadat podle výpisu 2.40.

Výpis č. 2.40

```
\usepackage[dvips]{color}
\definecolor{barvaodkaz}{rgb}{.1,.1,.4}
\usepackage[latin2]{inputenc}
\usepackage[dvips,bookmarksnumbered,unicode,colorlinks,%
linkcolor={barvaodkaz},citecolor={green},urlcolor={blue}]{hyperref}
```

Když máme načten balíček `hyperref`, odkazy „dovnitř dokumentu“ (obsah, odkazy na seznam literatury, na tabulky apod.) jsou označeny a aktivovány automaticky, odkazy na WWW stránky označíme například následujícími příkazy (druhý příkaz použijeme, pokud nechceme do dokumentu vložit celou adresu, ale odkaz má být funkční):

```
\url{http://nejaka.adresa.cz}
\href{http://nejaka.adresa.cz}{slovní odkaz}
```

2.11 Seznam literatury a rejstřík

Seznam literatury se vkládá na dané místo prostředím `thebibliography`, jehož parametrem je „nejširší číslo“ použité při číslování, obvykle se použije 99. Položky začínají příkazem `\bibitem`, kde jako povinný parametr zadáváme název pro křížové odkazy, jako nepovinný parametr můžeme použít slovní pojmenování položky (místo číslování). Ve výpisu 2.41 vidíme způsob vytvoření číslovaného seznamu literatury, za výpisem je vysázený seznam.

Výpis č. 2.41

```
\begin{thebibliography}{99}
\bibitem{linuxnoviny} \textsc{Dvořák}, V. \emph{WIN95 + RH6.2 = 10GB HDD.}
Linuxové noviny [$online$]. 2001\
URL: \url{http://www.linux.cz/noviny/2001-08/clanek05.html}
[$cit. 18. 10. 2006$]$
\bibitem{vaznyfeynman} \textsc{Feynman}, R. P. \emph{To snad nemyslíte vážně!}
Praha, Mladá Fronta, 1989.
\bibitem{libgaher} \textsc{Gahér}, F. \emph{Logické hádanky, hlavolamy
a paradoxy.} Bratislava, Iris, 1997.
\end{thebibliography}
```

- [1] DVOŘÁK, V. *WIN95 + RH6.2 = 10GB HDD*. Linuxové noviny [online]. 2001
URL: <http://www.linux.cz/noviny/2001-08/clanek05.html> [cit. 18. 10. 2006]
- [2] FEYNMAN, R. P. *To snad nemyslíte vážně!* Praha, Mladá Fronta, 1989.
- [3] GAHÉR, F. *Logické hádanky, hlavolamy a paradoxy*. Bratislava, Iris, 1997.

Na položky seznamu se odkazujeme příkazem `\cite[text]{odkaz}`. Povinný parametr `odkaz` je odkazem na položku, v nepovinném parametru můžeme zadat třeba číslo stránky nebo kapitoly. Podle výpisu 2.41 můžeme vytvořit třeba tyto odkazy:

```
\cite{linuxnoviny}
\cite[s. 25]{vaznyfeynman}
```

Položky seznamu literatury můžeme mít pojmenovány i slovně, obvykle se používá příjmení autora s rokem vydání, při více knihách jednoho autora ve stejném roce přidáváme ještě písmenka a, b, ... Použití vidíme na výpisu 2.42.

Výpis č. 2.42

```
\begin{thebibliography}{99}
...
\bibitem[Feynman89]{vaznyfeynman} \textsc{Feynman}, R. P. \emph{To snad nemyslíte
vážně!} Praha, Mladá Fronta, 1989.
...
\end{thebibliography}
```

Do seznamu literatury dostaneme:

...

[Feynman89] FEYNMAN, R. P. *To snad nemyslíte vážně!* Praha, Mladá Fronta, 1989.

...

Odkaz ve tvaru `\cite{vaznyfeynman}` se vysází jako [Feynman89], odkaz s nepovinným parametrem `\cite[s. 25]{vaznyfeynman}` jako [Feynman89, s. 25].

Pro vysázení *rejstříku* potřebujeme balíček `makeidx` nebo `csindex`. Ten první je určitě ve všech distribucích, ten druhý bude asi třeba doinstalovat.

Tedy načteme balíček, dále ještě v preambuli (třeba hned za příkazem pro načtení balíčku `makeidx`) napíšeme příkaz `\makeindex`, kterým dáme najevo, že se má vytvářet pomocný soubor pro vytvoření rejstříku. Na místě, kde chceme mít rejstřík vysázený, dáme příkaz `\printindex`.

Musíme také označit místa a pojmy, které chceme do rejstříku zahrnout. Na daném místě dokumentu pak použijeme příkaz `\index{pojem}`. Máme k dispozici i další možnosti strukturování rejstříku, najdeme je například v nápovědě. Užitečný může být také balíček `index`.

2.12 Dokumentové seznamy podle stylů

Vytváření dokumentových seznamů je v $\text{\LaTeX}$ u poměrně dobře automatizované. Technicky probíhá tak, že existují speciální soubory s tím, co má v daném seznamu být (s příponou `toc` pro obsah, `lot` pro seznam tabulek, `lof` pro seznam obrázků, `idx` pro rejstřík, apod.), a při překladu se údaje berou právě z těchto souborů.

Seznamy se vytvářejí z toho, co bylo vygenerováno při minulém překladu. Proto je nutné při změně údajů v souborech (třeba přidání kapitoly) *provést překlad alespoň dvakrát*, aby seznamy byly aktuální, případně i vícekrát (když se vložením nebo rozšířením seznamu mění rozložení stránek, například obsah se rozšíří na další stránku).

Obsah, seznam tabulek a seznam obrázků vložíme na dané místo příkazy

```
\tableofcontents
\listoftables
\listoffigures
```

Dokonce tyto příkazy mohou následovat přímo za sebou (nemusíme mít samozřejmě všechny, jen ty, které potřebujeme), ale pokud používáme třídu `article` nebo z ní odvozenou, je vhodné mezi ně vložit příkaz `\newpage`.

Do obsahu se automaticky neřadí nečíslované nadpisy. Protože však občas potřebujeme mít v obsahu kapitolu, kterou nechceme číslovat, nebo dokonce chceme přidat řádek, který se ke kapitolám vůbec nevztahuje, máme k dispozici dva příkazy, a to příkaz `\addcontentsline`, který používáme při vložení „běžného“ nadpisu do určité úrovně obsahu (zadááme úroveň slovem `chapter`, `section`, `subsection`

apod.), a příkaz `\addtocontents` sloužící ke vložení prakticky jakéhokoliv textu do obsahu (můžeme také používat formátovací příkazy, ale ne úplně libovolně).

Výpis č. 2.43

```
\chapter*{Úvod}
\addcontentsline{toc}{chapter}{Úvod}
...
\chapter*{Přílohy}    % nadpis můžeme i formátovat nebo oživit obrázkem, což jsme
\addtocontents{\bigskip\noindent\bfseries{Přílohy}}           % zde nepoužili
\thispagestyle{empty}
...
\newpage
\addcontentsline{toc}{chapter}{Rejstřík}
\printindex
```

Ve výpisu 2.43 máme nejdřív vložení nečíslované kapitoly s názvem Úvod do obsahu (zkratka `toc`) na první úroveň (`chapter`, ve třídě `article` bychom zde použili `section`). Dále vkládáme do obsahu řádek oddělující přílohy od běžných kapitol, ale nechceme na tomto řádku obsahu číslo stránky, které jsme ostatně odbourali i na stránce samotné, proto použijeme příkaz `\addtocontents` a sami si určíme, co a jak bude zobrazeno (inspiraci můžeme hledat přímo v souboru s příponou `toc` stejně nazvaném jako samotný dokument).

Třetí zásah do obsahu je vložení odkazu na kapitolu s rejstříkem. Tato kapitola se sice v dokumentu správně nazve, ale není zahrnuta do rejstříku. Nemůžeme použít přesně stejný postup jako u kapitoly Úvod, protože při použití příkazu `\addcontentsline` až za příkazem načítajícím rejstřík by se do obsahu vložilo číslo poslední strany rejstříku místo strany s jeho začátkem, proto příkaz umístíme před příkaz pro načtení rejstříku, ale aby se nevložilo číslo o 1 menší než chceme, musíme ještě použít příkaz `\newpage`.

Zatím jsme v příkazech `\addcontentsline` a `\addtocontents` používali zkratku `toc`, která znamenala, že pracujeme s obsahem. Tyto příkazy můžeme použít také na seznam tabulek a seznam obrázků, ale místo `toc` musíme použít `lot` (tabulky) nebo `lof` (obrázky).

2.13 Rozměry stránky

Protože evropské dokumenty mají obvykle poněkud jiné rozměry potištěné oblasti stránky než americké (T_EX je americký produkt), můžeme použít balíček `a4wide`, který nastaví evropské rozměry podle formátu A4, ale u třídy dokumentu musíme mít uveden nepovinný parametr `a4paper`, například takto:

```
\documentclass[a4paper,12pt]{article}
\usepackage{czech,a4wide}
```

Jestliže potřebujeme upravit rozměr (délku) pouze jedné stránky, vystačíme si s příkazem `\enlargethispage{vzdálenost}`, kde vzdálenost je délka, o kterou

chceme stránku prodloužit (nebo zkrátit, když použijeme záporné číslo). Například pokud se nám nevejde na stránku něco, co nechceme mít posunuto až na další stránku, použijeme tento příkaz s parametrem 1em (samozřejmě na té stránce, kterou chceme prodloužit, raději výše). Když to nestačí, zvážíme, zda opravdu nechceme daný objekt nebo odstavec umístit až na další stránce, měli bychom brát v úvahu i estetické hledisko.

Veškeré rozměry týkající se stránky jsou uloženy v délkových registrech. Některé z nich můžeme měnit, jiné ne, obvykle z důvodu vzájemné závislosti. Více se podíváme na tuto problematiku až v příloze B.2, teď si ukážeme na příkladu způsob nastavení některých rozměrů stránky.

U některých údajů najdeme záporná čísla. Je to proto, že u horního a levého okraje se ještě navíc počítá „prázdná“ vzdálenost přibližně 1 palec (1in), kterou můžeme případně záporným číslem zmenšit. Například horní (prázdný) okraj nad textem je 1 palec plus (nebo minus) obsah určitého délkového registru.

Příklad 2.20 Použijeme délkové registry pro přímé nastavení rozměrů stránky. Všechny níže uvedené příkazy musí být v preambuli dokumentu, tedy ještě před příkazem `\begin{document}`.

Výpis č. 2.44

```
\setlength{\topmargin}{-0.2in} % horní okraj
\setlength{\topskip}{0.3in} % mezi záhlavím a textem
\setlength{\textheight}{9.6in} % výška textu
\setlength{\leftmargin}{1cm} % (+1in) levý okraj
\setlength{\textwidth}{6in} % šířka textu
```

Z těchto registrů už známe příkaz `\textwidth`, který jsme používali pro zjištění šířky potiskované části stránky, ostatní už nejsou z tohoto hlediska tak důležité, ale také užitečné. Například když chceme vynechat půl strany místa (vertikálně), použijeme tento příkaz:

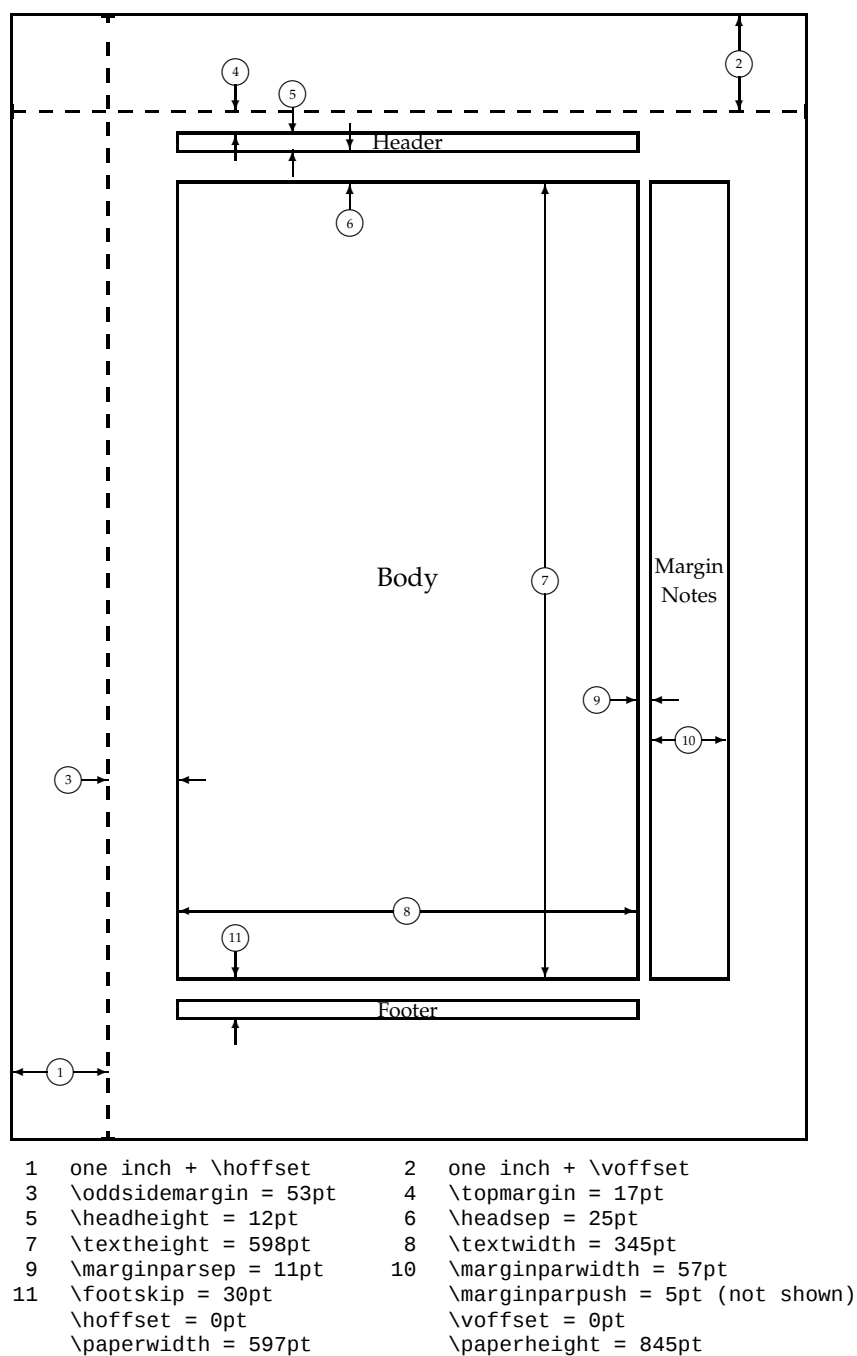
```
nějaký text
\vspace{.5\textheight}

další text po vynechané polovině strany
```

Délkové registry nemusíme nastavovat přímou hodnotou, můžeme jen upravovat o určitou relativní hodnotu. Například když chceme rozšířit textovou část o jeden centimetr, napíšeme příkaz

```
\addtolength{\textwidth}{1cm}
```

Různé možnosti nastavení rozměrů stránky máme v balíčku `geometry`. Užitečný může být také balíček `layout`, ve kterém je definován příkaz `\layout`, ukázkou jeho výstupu vidíme na obrázku 2.9 na straně 92.

Obrázek 2.9: Vykreslení rozměrů stránky příkazem `\layout`

2.14 Píšeme pro vědu

Tato kapitola není určena jen matematikům, ale všem, kdo píšou dokument s větami, důkazy, příklady či jinými podobnými prostředními (to se týká téměř všech vědních oborů), vytvářejí struktury či diagramy (také třeba lingvisté) nebo píšou postupy (algoritmy).

2.14.1 Definice, věty a důkazy

Exaktní věda je postavena na logice a logickém deduktivním vyvozování. To znamená, že v každém vědním oboru máme základní pojmy a základní (odpovězované) vztahy, a pak všechno ostatní – další pojmy, které definujeme pomocí základních nebo již definovaných pojmů, a odvozené vztahy (věty, teoremy, lemmata), které jsme odvodili pomocí základních nebo již odvozených vztahů.

Pokud se pohybujeme i v teoretické části některého vědního oboru, potřebujeme občas psát vlastní definice pojmů (i dříve definovaných, pro připomenutí) a také vlastní věty, teoremy („důležitější“, nové věty), lemmata (pomocné věty) a zapisovat jejich důkazy. Dalším prvkem z této oblasti jsou příklady, ukázky, úlohy a poznámky vysvětlující či doplňující věty a definice.

Pro zápis těchto prvků existují určité konvence. *Definice, různé druhy vět a poznámky* se sázejí kurzívou, vyznačuje se v nich zrušením kurzívy (tedy v $\text{\LaTeX}$ u prostě použijeme příkaz `\emph`), jsou číslovány. V článku číslování nezávisí na čísle kapitoly, ale v rozsáhlejšímu textu (třeba diplomové práci) ano.

Důkazy se nečíslojí, přísluší vždy k určité větě a hned za ní bývají uváděny (pokud z nějakého důvodu musí být dál, třeba když za větu chceme vložit lemma s důkazem ulehčující důkaz té věty, bývá za označením důkazu v hranatých závorkách odkaz na větu, kterou dokazujeme). Důkazy již nesázíme kurzívou, ale normálním písmem.

Příklady, ukázky či úkoly bývají číslovány, pro číslování platí totéž co pro věty, a stejně jako u důkazů používáme normální řez písma.

Pro číslování se používají čítače, jakési proměnné, které se při každém použití zvýší o 1. Různé prvky mohou mít různé čítače, ale často se setkáváme s tím, že třeba všechny druhy vět používají tentýž čítač.

$\text{\LaTeX}$ nám dává možnost vytvářet si svá vlastní prostředí s takovým označením a číslováním, jaké budeme chtít. Tuto problematiku jsme už trochu nakousli v kapitolách 2.6.2 o stylech a 2.8 o křížových odkazech.

Nové prostředí tohoto typu vytvoříme pomocí příkazu `\newtheorem`, kde zadáváme titulek prostředí a určujeme způsob číslování, které může být závislé na čísle kapitoly, případně lze číslování propojit s jiným takovým prostředím. Ukázku máme ve výpisu 2.45.

Výpis č. 2.45

```
\newtheorem{veta}{Věta}[section] % prostředí závislé na číslování kapitol section
\newtheorem{lemma}[veta]{Lemma} % prostředí číslované zároveň s prostředím veta
\newtheorem{pozorovani}{Pozorování}[section]
\newtheorem{definice}{Definice}[section] % prostředí pro definice
\newtheorem{thpriklad}{Příklad} % „meziprodukt“, použito dále:
\newenvironment{priklad}%
  {\begin{thpriklad}\normalfont}%
  {\end{thpriklad}}
\newenvironment{dukaz}%
  {\textbf{Důkaz:}\quad}%
  {\$Box$} % pro tento příkaz je nutno načíst balíček latexsym nebo amssymb
```

Ukážeme si použití prostředí definovaných ve výpisu 2.45. Všimněte si ukončení prostředí pro důkazy – prázdný čtvereček se pro tyto účely používá dost často, má

zde význam „hotovo“.

Výpis č. 2.46

```
\begin{definice} Říkáme, že sněží, pokud
padají sněhové vločky.
\end{definice}
```

```
\begin{definice} Sníh je hmota složená
ze sněhových vloček.
\end{definice}
```

```
\begin{pozorovani}\label{poz:snihmraky}
Sněhové vločky padají z~tmavě
šedých mraků.
\end{pozorovani}
```

```
\begin{lemma} Když padá sníh, jsou na
obloze šedé mraky.
\end{lemma}
```

```
\begin{dukaz} Plyne z~Pozorování
\ref{poz:snihmraky}).
\end{dukaz}
```

```
\begin{veta} Když venku sněží, je
teplota menší než 5~\celsius.
\end{veta}
```

```
\begin{dukaz} Sněhové vločky se pohybují
vzduchem. Pokud je teplota blízko země
5~\celsius nebo vyšší, vločky roztávají
ještě ve vzduchu a~padá déšť.
\end{dukaz}
```

```
\begin{priklad} Podíváme se na oblohu.
Co tam vidíme? Pokud šedé mraky, můžeme
očekávat sněžení.
\end{priklad}
```

Definice 2.1 Říkáme, že sněží, pokud padají sněhové vločky.

Definice 2.2 Sníh je hmota složená ze sněhových vloček.

Pozorování 2.1 Sněhové vločky padají z tmavě šedých mraků.

Lemma 2.1 Když padá sníh, jsou na obloze šedé mraky.

Důkaz: Plyne z Pozorování 2.1. $\square$

Věta 2.2 Když venku sněží, je teplota menší než 5 °C.

Důkaz: Sněhové vločky se pohybují vzduchem. Pokud je teplota blízko země 5 °C nebo vyšší, vločky roztávají ještě ve vzduchu a padá déšť. $\square$

Příklad 1 Podíváme se na oblohu. Co tam vidíme? Pokud šedé mraky, můžeme očekávat sněžení.

Existují balíčky, které rozšiřují naše možnosti, například `ntheorem`. Dále v téměř každém balíčku se třídami nebo styly pro odborné články nebo rozsáhlejší práce bývají tato prostředí předdefinována.

2.14.2 Vzorce nejen matematické

Matematici, fyzikové, chemici, ekonomové a další obyvatelé této planety občas potřebují zapsat vzorec či více zarovnaných vzorců, a to buď samostatně na řádku (případně s číslováním), nebo v odstavci s textem.

Jednoduché vložení takového výrazu do textu jsme už vlastně používali – například výraz $x = y + 2/5$ vysázíme sekvencí `$x=y+2/5$`. Když tuto sekvenci ohraničíme místo symbolů `$` jejich dvojicemi (tedy `$$x=y+2/5$$`), vysází se výraz na samostatný

řádek.

Výrazy na samostatném řádku by správně měly být ve většině případů číslovány, aby bylo možné na ně odkazovat. Pak místo symbolů \$ použijeme matematická prostředí `equation` (jeden řádek) nebo `eqnarray` (i více řádků s možností určit zarovnání pomocí symbolu `&`, viz příklad 2.21, ale nemusíme použít). Na řádky můžeme přidat příkaz `\label` a pak jeho argument používat pro získání čísla rovnice. Když určitý řádek v soustavě nechceme mít číslováný, použijeme na tomto řádku příkaz `\nonumber`.

Pro samotný obsah těchto matematických úseků nebo prostředí využíváme nejen matematické operátory, ale také funkce, speciální symboly a případně i řeckou abecedu. To nejzákladnější najdeme obvykle v menu nebo panelech nástrojů editoru, u editoru $\text{\TeX}$ nicCenter buď na jednom panelu nástrojů nebo v menu Insert $\rightarrow$ Formulas (ohraničení matematických prostředí) a Moth (všechno ostatní).

Příklad 2.21 Ukážeme si zápis číslované rovnice a soustavy rovnic, a také odkaz na rovnici. Soustava rovnic bude zarovnána podle operátorů rovnosti a nerovnosti. Všimněte si způsobu zarovnání pomocí `&` v prostředí `eqnarray`.

Výpis č. 2.47

Rovnice:

```
\begin{equation}
x=y+2/5\label{eq:rovnice}
\end{equation}
```

To byla rovnice (`\ref{eq:rovnice}`).

Soustava:

```
\begin{eqnarray}
y-5+\frac{-x}{3} & \leq & \sqrt{2}*x \\
x/2+y & < & 0 \\
\sin(x)+1 & \geq & \cos(y) \\
x^2+y^{2n}-x^{2^t} & = & vysl_3 \\
vysl_2 & = & \sum_{i=1}^n x^i
\end{eqnarray}
```

První rovnice soustavy je (`\ref{eq:deleniodmocninou}`).

Rovnice:

$$x = y + 2/5 \quad (2.1)$$

To byla rovnice (2.1).

Soustava:

$$y - 5 + \frac{-x}{3} \leq \sqrt{2} * x \quad (2.2)$$

$$x/2 + y < 0$$

$$\sin(x) + 1 \geq \cos(y) \quad (2.3)$$

$$x^2 + y^{2n} - x^{2^t} = vysl_3 \quad (2.4)$$

$$vysl_2 = \sum_{i=1}^n x^i$$

První rovnice soustavy je (2.2).

To, že máme k dispozici spoustu různých symbolů z různých vědních oborů, už víme (viz strana 50 v kapitole 2.4.4), ale podpora pro tyto účely je v $\text{\LaTeX}$ u mnohem širší. Ve třídách a stylech pro specializované odborné časopisy a jiné publikace najdeme mnohá užitečná nastavení, prostředí a příkazy, a také existuje mnoho balíčků se styly pro různé účely. Například balíček `xymtex` je určen pro vytváření formulí s chemickými strukturami, `tensor` obsahuje podporu sázení tenzorů s horními i dolními indexy správně zarovnanými, `xlop` dokáže nejen formátovat, ale také provádět matematické výpočty, atd.

2.14.3 Stromové struktury, diagramy a grafy

Pro strukturovaná data samozřejmě můžeme použít prostředí `picture` nebo některý balíček, který rozšiřuje jeho možnosti, ale máme také k dispozici balíčky určené přímo pro tyto účely.

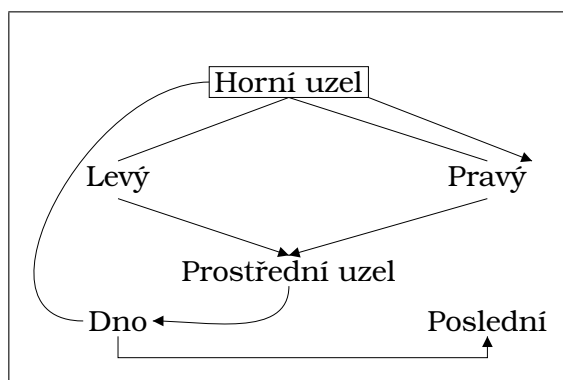
Stromové (nebo jakkoliv rozvětvené) struktury můžeme vytvořit pomocí různých balíčků, například `tree-dvips`, `xyling`, `trees`, `qtree`, `ps-trees`, atd. Většina z nich používá pro vykreslování čar a křivek Postscript, proto je třeba formát PDF tvořit přes formát PS (v T_EXnicCenter profil překladu LaTeX=>PS=>PDF).

Příklad 2.22 Ve výpisu a na výsledku máme ukázkou použití balíčku `tree-dvips`. Vykreslíme celkem šest uzlů, které jsou nazvány `x`, `y`, `z`, `u`, `v`, `p` (na tyto názvy se odkazujeme v dalších příkazech) a každý má svůj text (například uzel `x` má text Horní uzel). Uzly umístíme pomocí tabulky.

Pak propojíme uzly čarami, šipkami, křivkami (zde musíme zadat, jak moc bude „vykřivená“) a křivkami se šipkou. U všech těchto prvků můžeme v nepovinných parametrech určit body, ze kterých a do kterých čára nebo křivka povede. Například písmenko `t` určuje horní část uzlu, `bl` spodní levou (bottom-left) část.

Výpis č. 2.48

```
% Nejprve vytvoříme a umístíme uzly:
\begin{tabular}{ccc}
& \node{x}{Horní uzel}\l[2em]
& \\
\node{y}{Levý}&&\node{z}{Pravý}\l[2em]
& \node{u}{Prostřední uzel}\l[1ex]
\node{v}{Dno}&&\node{p}{Poslední}
\end{tabular}
% Orámuje první uzel:
\nodebox{x}
% Nakreslíme čáry:
\nodeconnect{x}{y}
\nodeconnect{x}{z}
% Nakreslíme šipky:
\anodeconnect{y}{u}
\anodeconnect{z}{u}
\anodeconnect[br]{x}{tr}{z}
% Nakreslíme křivku a křivku s šipkou:
\nodecurve[l]{x}{l}{v}{1.6cm}
\nodecurve[b]{u}{r}{v}{.7cm}
% „Rovná“ šipka dole:
\abarconnect[-8pt]{v}{p}
```



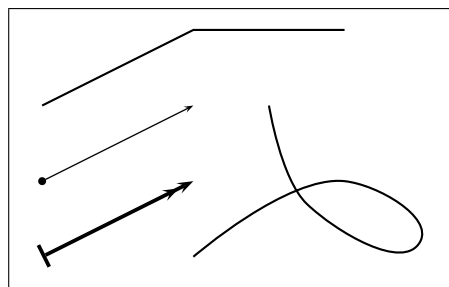
Dále si ukážeme použití některých příkazů balíčku `pstricks`. Jak název napovídá, opět budeme překládat přes formát PS. Samotný balíček nabízí spoustu příkazů souvisejících s grafikou, my si zde ukážeme jen to nejzákladnější – čáry s různými zakončeními a křivku. Tento balíček nelze používat zároveň s balíčkem `color`, což ale nevadí, protože má vlastní (a dokonalejší) správu barev. Máme k dispozici prostředí `pspicture` podobné již známému prostředí `picture`, předdefinované jednotky jsou však centimetry, ne body.

Výpis č. 2.49

```

\begin{pspicture}(6,3)
% Lomená čára:
\psline(0,2)(2,3)(4,3)
% Šipky:
\psline[arrows=*->,linewidth=.5pt](0,1)(2,2)
\psline[arrows|=>>,linewidth=.5mm](0,0)(2,1)
% Křivka:
\pscurve(2,0)(4,1)(5,.2)(3.5,.7)(3,2)
\end{pspicture}

```



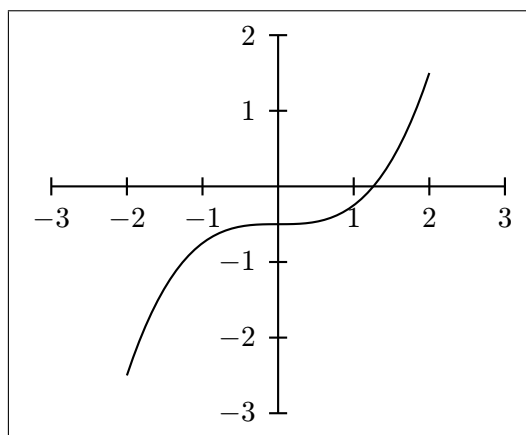
Další možnosti nám poskytne kombinace balíčku pstricks a některého dalšího, například ps-plot (tj. příkazem `\usepackage` načteme nejdřív pstricks a pak ještě ps-plot), v našem případě máme k dispozici kromě jiného příkazy k vykreslování funkcí:

Výpis č. 2.50

```

\begin{pspicture}(0,0)(4,5)
% Soustava souřadnic:
\psaxes(0,0)(-3,-3)(3,2)
% Funkce:
\psplot{-2}{2}{x^3 exp 4 div 0.5 sub}
\end{pspicture}(6,3)

```



První příkaz vykreslí soustavu souřadnic, druhý příkaz funkci $x^3/4 - 0.5$ na intervalu $x \in (-2, 2)$ (to jsou první dva parametry)¹².

Pro vkládání grafů a jiných složitých nákrešů opět existuje více možností. Některé balíčky dokážou vykreslit jednoduché funkce přímo (to jsme viděli u pstricks + ps-plot), pro složitější nebo vícerozměrné funkce je lepší využít specializované nástroje. Máme dvě základní možnosti:

- a) použijeme kterýkoliv program pracující s grafikou nebo přímo s grafy a uložíme či exportujeme obrázek do vhodného formátu (většina těchto programů zvládá

¹²Zápis funkce je v postfixu, kdy se nejdřív píšou oba operátory a pak název funkce nebo operátor, například $4 + 5$ se запиše jako $4\ 5\ +$ nebo výraz $2 * x + 4$ jako $2\ x\ *\ 4\ +$.

Postfixový zápis vytvoříme jednoduše: nejdřív si „uzávorkujeme“ celý výraz, například $a + 2 * x + 4 - z$ uzavřeme na $((a + (2 * x)) + 4) - z$, postupujeme podle toho, které operátory mají přednost, v případě několika se stejnou předností (třeba +, -) postupujeme zleva. Pak jdeme od nejvnitřnějších závorek – nejdřív napíšeme $2\ x\ *$, pak řešíme $+$ s výsledkem $a\ 2\ x\ *$ + (operátor přidáváme na konec, operandy podle pořadí v původním výrazu), atd., v našem případě bude výsledek $a\ 2\ x\ *\ 4\ +\ z\ -$.

export do EPS, PDF nebo něčeho převeditelného na tyto formáty) – *Mathematica*, *Inkscape* ([P20]), *Corel*, apod., výborným a přitom volně šiřitelným programem speciálně pro tyto účely navrženým je *GNUPlot* ([P21], pěkný úvod na [P22]),

- b) existují programy vytvářející přímo T_EXový kód – *LT_EX_{Pi}X* ([P23]), *T_EXcad32* ([P24]), *T_EXCAD* ([P25]), atd.

2.14.4 Postupy, algoritmy, předem zformátovaný text

V odbornějších publikacích včetně diplomových prací je někdy třeba vložit *strukturovaný postup* nějaké činnosti. Nejde jen o algoritmus při programování, ale obecně o jakýkoliv postup nebo způsob konstrukce, který chceme napsat co nejpřehledněji. L^AT_EX nám opět nabízí několik balíčků, z nichž jsou zajímavé například *algorithm2e*, *algorithmicx*, *pseudocode*, *newalg*. Některé z nich, například *algorithm2e*, nabízejí možnost vygenerovat seznam algoritmů podobně jako generujeme třeba obsah nebo seznam obrázků (příkaz je `\listofalgorithms`, v případě uvedeného balíčku má i podporu češtiny).

Příklad 2.23 Ukážeme si základní možnosti balíčku *alorighm2e*. Balíček načteme následujícím příkazem (v preambuli) s nepovinnými parametry pro vzhled a nastavení češtiny v popiscích a seznamu algoritmů:

```
\usepackage[ruled,vlined,czech]{algorithm2e}
```

Dále vytvoříme dva algoritmy. První popisuje postup při přecházení křižovatky (bez světelné signalizace), druhý postup čtení detektivky. U druhého postupu budeme chtít očíslovat řádky.

Výpis č. 2.51

```
\begin{algorithm}[H]
příchod ke křižovatce\;
\Repeat{nejede auto}{
  pohlédni vlevo\;
  \While{jede auto}{čekej\;
  pohlédni vpravo\;
  \While{jede auto}{čekej\;
  pohlédni vlevo\;
}
přejdi\;
\caption{Přechod křižovatky}
\label{alg:krizovatka}
\end{algorithm}
```

Algoritmus 1: Přechod křižovatky

```
příchod ke křižovatce;
repeat
  pohlédni vlevo;
  while jede auto do
    _ čekej;
  pohlédni vpravo;
  while jede auto do
    _ čekej;
  pohlédni vlevo;
until nejede auto ;
přejdi;
```

Výpis č. 2.52

```

\linesnumbered
\begin{algorithm}[H]
vezmi knihu\;
otevři knihu na první straně\;
\While{není konec knihy}{
  čti kapitolu\;
  \eIf{rozumíš kapitole}{
    přesuň se na další kapitolu\;
    \If{víš kdo je vrah}{
      přestaň číst\;
      přesuň se na konec knihy\;
    }
  }{
    zpět na začátek kapitoly\;
  }
}
\caption{Hledání vraha}
\label{alg:hledanivraha}
\end{algorithm}

```

Algoritmus 2: Hledání vraha

```

1 vezmi knihu;
2 otevři knihu na první straně;
3 while není konec knihy do
4   čti kapitolu;
5   if rozumíš kapitole then
6     přesuň se na další kapitolu;
7     if víš kdo je vrah then
8       přestaň číst;
9       přesuň se na konec knihy;
10  else
11    zpět na začátek kapitoly;

```

Jak vidíme, vlastně jde o nový typ plovoucích prostředí. Pokud se prostředí nevejde celé na stránku, kde je zadáno, vysází se až na další stránku, „poplave“ v textu o něco dál. Pokud tomuto chování chceme zabránit (nebo při umístění do prostředí minipage), použijeme nepovinný parametr H.

K postupům charakteru „programovacího“ nebo výpisům textových souborů patří možnost vložit text „tak jak je“, tedy včetně případných speciálních znaků, na které by $\text{\LaTeX}$ nějakým způsobem reagoval, mezer a konců řádků. K tomu slouží buď příkaz `\verb` nebo prostředí verbatim či alltt.

Příkaz `\verb` má trochu zvláštní použití. To, co chceme pomocí tohoto příkazu vložit, ohraničíme jakýmkoliv symbolem, který se v ohraničovaném textu nenachází (stejný symbol před i za textem, tedy ne levá a pravá závorka!), například:

Výpis č. 2.53

```

\verb+http://www.adresa.cz+\\
příkaz \verb-\verb-\\
soubor \verb|C:\dokumenty\zivotopis.txt|

```

```

http://www.adresa.cz
příkaz \verb
soubor
C:\dokumenty\zivotopis.txt

```

Prostředí verbatim pracuje jako jiná prostředí, tedy text, který chceme nechat zachovaný tak, jak je (včetně konců řádků apod.), umístíme mezi `\begin{verbatim}` a `\end{verbatim}`.

Prostředí alltt z balíčku alltt (tento balíček musíme samozřejmě načíst) je svou funkcí podobné verbatim, ale navíc interpretuje některé příkazy $\text{\LaTeX}$ u. Toto prostředí používáme, pokud chceme zachovávat mezery a konce řádků, ale navíc používáme barvy, řezy a další příkazy.

Úlohy:

1. Vytvořte nový dokument třídy `article` se základní velikostí písma 10 bodů.
 2. Do preambule dokumentu napište tento příkaz (vytvoření příkazu pro kapitolu): `\newcommand{\kapitola}[1]{\section{#1}}`
 3. Nadefinujte si podobný příkaz pro podkapitolu:
`\podkapitola{Název podkapitoly}`
odkazující pro změnu na příkaz `\subsection` s předáním parametru.
 4. Oba tyto příkazy pro kapitolu a podkapitolu použijte v dokumentu (překládejte do DVI), přidejte také běžný text, v něm některá slova vyznačte *vhodným* řezem.
 5. Změňte třídu dokumentu na `report` a dále v definicích příkazů `\kapitola` a `\podkapitola` změňte odkazy na příkazy `\chapter` a `\section` (opět s předáním parametrů), aby nadpisy odpovídaly použití v této třídě.
 6. Vytvořte a umístěte titulní stranu. Zvolte nadpis a sebe napište jako autora.
 7. Na konci dokumentu vytvořte seznam literatury s alespoň dvěma položkami (vhodně formátovanými). Do první kapitoly umístěte citaci některé z položek seznamu literatury.
 8. V některé z kapitol vytvořte jednoduchý obrázek (náčrt – pomocí prostředí `picture`), přidejte titulek (tj. vložte tento obrázek do příslušného plovoucího prostředí).
 9. V některé z kapitol vytvořte jednoduchou tabulku s obsahem podle vlastního výběru (alespoň dva sloupce a alespoň dva řádky), orámování si zvolte sami. Tabulku umístěte do příslušného plovoucího prostředí.
 10. Za titulní stranu (tj. před kapitoly) umístěte *obsah*, *seznam tabulek* a *seznam obrázků*.
 11. Z tohoto dokumentu vytvořte PDF soubor.
-

KAPITOLA 3

OpenOffice.org

V této kapitole najdeme informace o kancelářském balíku, který se pro mnohé stal alternativou MS Office, především o v něm obsaženém textovém editoru Writer, obdobě MS Wordu.

Zaměříme se především na to, co je ve Writeru jiné oproti MS Wordu, čehož není až tak hodně, proto tato kapitola bude mnohem kratší než předchozí kapitoly věnované MS Wordu a $\text{\LaTeX}$ u.

3.1 Kancelářský balík OpenOffice.org

Kancelářský balík OpenOffice.org¹ je považován za zajímavou a prakticky plnohodnotnou alternativu balíku MS Office. V současné době (začátek roku 2007) je dostupný ve verzi 2.1, což je výrazně nižší číslo než v případě MS Office (MS Office XP mělo vnitřní verzi 10, MS Office 2003 má verzi 11, byla vydána beta-verze MS Office 2007, které samozřejmě mají verzi ještě vyšší), tento produkt měl na vývoj mnohem kratší dobu než konkurent.

V OpenOffice.org najdeme tyto programy:

Writer – textový editor přibližně odpovídající MS Wordu, narozdíl od něho je však více zaměřen na používání stylů, od verze 2.0 používá jako výchozí formát ODT.

Calc – tabulkový procesor přibližně odpovídající MS Excelu, funkce jsou srovnatelné, od verze 2.0 používá formát ODS.

Impress – nástroj pro tvorbu prezentací přibližně odpovídající MS PowerPointu, podobný je i styl práce, jen šablon je o něco méně, výchozím formátem od verze 2.0 je ODP.

¹Přípona .org je opravdu součástí názvu, nejde jen o WWW adresu (i když ta existuje také v tomto tvaru).

Draw – grafický (vektorový) editor, styl práce je trochu podobný tomu, co známe z vytváření nákrešů v MS Office, ale máme více možností včetně práce s vrstvami, formátem od verze 2.0 je ODG.

Base – jednoduchý databázový systém trochu podobný MS Accessu, od verze 2.0 používá formát ODB.

Math – nástroj pro tvorbu rovnic přibližně odpovídající obdobnému nástroji dostupnému v MS Office, formát je příbuzný $\LaTeX$ u, i když pracujeme „vizuálně“, formátem od verze 2.0 je ODF.

Jak je z výčtu patrné, od verze 2.0 tohoto balíku jsou používány formáty souborů ODT (text), ODS (tabulky), ODP (prezentace), ODG (grafika), ODB (databáze) a ODF (matematika, formule). První dvě písmena těchto formátů jsou zkratkou slov *Open Document*. Všechny patří do skupiny formátů nazvané *Open Document Format* (ODF), což jsou otevřené textové formáty (textové ve smyslu čitelnosti souborů také v jednoduchých textových editorech, formátování je prováděno textovými příkazy). Tyto formáty se pravděpodobně budou používat v budoucnosti velmi často, a to i na úradech. Další formáty z této jsou ODC (grafy), ODI (obrázky), ODM (hlavní textový dokument) a obdobné pro šablony (prostřední písmeno je T místo D).

Jak vidíme, vybavenost programy je obdobná jeho konkurentovi MS Office. Podíváme se na srovnání dalších vlastností:

1. *Cena a dostupnost* je asi nejdůležitějším ukazatelem při rozhodování mnoha uživatelů – zatímco cena MS Office se pohybuje od několika tisíc až přibližně k 18 000 Kč (nejvybavenější varianta MS Office 2003), OpenOffice.org je zdarma ke stažení, a to přímo v nejvybavenější variantě.
2. *Přenositelnost* je výhodou na straně OpenOffice.org. Tento balík existuje ve variantách pro všechny nejrozšířenější operační systémy (MS Windows, Linux, různé Unixy, MacOS), zatímco MS Word je pouze pro MS Windows a v omezené míře pro MacOS.
3. *Přenositelnost formátů* mezi kancelářskými programy je opět lepší u formátů OpenOffice.org. Soubory v těchto formátech otevřeme nejen v původním balíku, ale také v celé řadě dalších (seznam je například na [O47]), a to naprosto bez problémů (je to otevřený formát s dobře čitelnou specifikací). Uzavřené formáty MS Office jen v málokterých aplikacích (mimo jiné i v OpenOffice.org), navíc s problémy u „složitějších“ dokumentů a ještě k tomu pouze díky mnoha nadšencům, kteří dlouhé dny, týdny, měsíce a roky experimentovali s programy patřícími do MS Office.
4. *Export do PDF* zvládají programy OpenOffice.org od verze 1.1 v poměrně dobré kvalitě (dokonce i prezentace), u MS Office potřebujeme externí programy.
5. *Ve vybavenosti* šablonami a dalšími předdefinovanými částmi vede zatím MS Office.

Určitým problémem se může zdát „dualita formátů“ pro různé verze OpenOffice.org, před verzí 2.0 byly totiž používány formáty SXW (text), SXC (tabulky), SXI

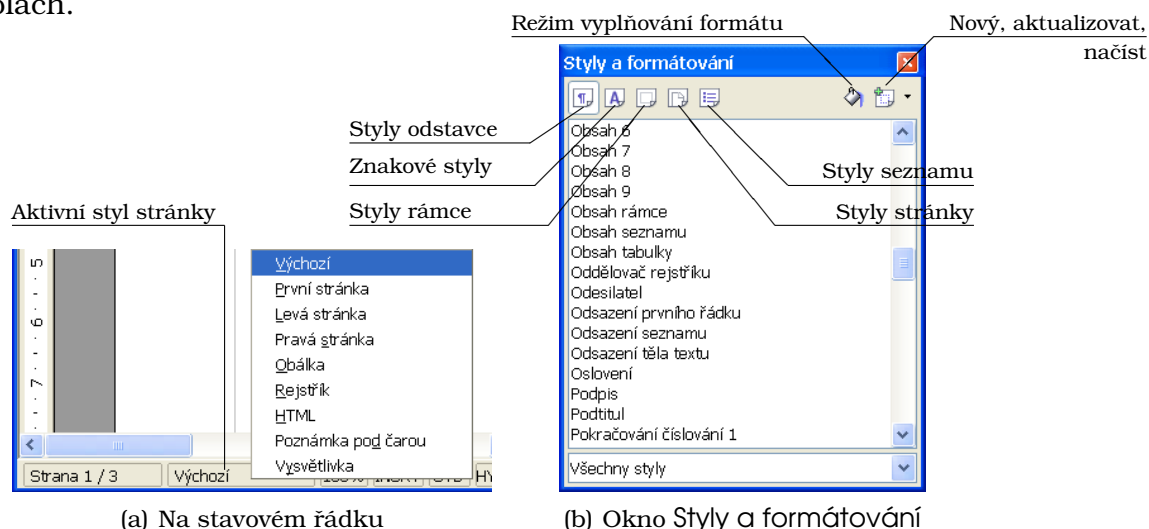
(prezentace) atd. Existují však konverzní programy, které dokážou soubor ve starším formátu převést do formátu nového, a navíc narozdíl od MS Office můžeme mít na jednom počítači nainstalovanou starší i novější verzi (třeba 1.1 a 2.1), dokonce je možné mít obě spuštěné zároveň.

Dále se budeme zabývat především programem *Writer*, který v OpenOffice.org slouží k tvorbě textových dokumentů. Způsob práce v ostatních částech balíku je podobný (především používáme styly), samozřejmě až na odlišnosti plynoucí z různého určení programů, proto bychom například při importování tabulky nebo grafu vytvořeného v Calcu či vektorového obrázku z Draw neměli mít problémy.

3.2 Práce se styly

Writer, stejně jako ostatní části OpenOffice.org, „tlačí“ uživatele k používání stylů. Typickým postupem tvorby dokumentu by mělo být (stejně jako u $\text{\LaTeX}$ u) vybrání a případná úprava stylů a pak jen psaní (případně vkládání objektů) a formátování těmito styly.

Ve Writeru rozlišujeme *styl znaku* (styly pro písmo), *odstavce*, *rámce* (styly pro umístění obrázků, tabulek apod.), *stránky* (záhlaví, zápatí, číslování apod.) a *seznamu* (číslované a odrážkové seznamy). Každý z těchto druhů stylů má své specifické vlastnosti, jejich nastavováním se budeme zabývat v samostatných podkapitolách.



Obrázek 3.1: Styly v OpenOffice.org

Ke stylům odstavce se dostaneme nejrychleji na hlavním panelu, stejně jako v MS Wordu. Tam vybíráme styly pro text, který píšeme, a také máme možnost

Nejrychlejší přístup ke stylům stránky máme ze *stavového řádku* (spodní okraj okna, viz obrázek 3.1(a)). Je zde zobrazen právě aktivní styl (na začátku *Výchozí*). Klepnutím pravým tlačítkem myši na této části stavového řádku získáme seznam stylů stránky, ze kterého můžeme přímo vybrat, poklepáním na toto místo dostaneme možnost měnit vlastnosti vybraného stylu.

Okno se všemi styly Styly a formátování zobrazíme pomocí menu Formát → Styly a formátování nebo klávesou . Toto okno si také můžeme přichytit k pravé straně hlavního okna (doporučuji; dělá se to tak, že myši „chytíme“ za titulek okna a táhneme směrem doprava až k okraji).

Na horním okraji tohoto okna je řada tlačítek, jsou popsána na obrázku 3.1(b) na straně 103. Tlačítka vlevo určujeme, které styly mají být zobrazeny, další tlačítko zapíná/vypíná *režim vyplňování formátu* (viz následující odstavec) a poslední nabízí menu pro práci s vybraným stylem (prakticky totéž získáme v kontextovém menu samotného stylu) – můžeme *vytvořit nový styl* vycházející z právě vybraného, *aktualizovat vybraný styl* (v dokumentu klepneme na text či objekt, který se má stát předlohou pro aktualizaci, a pak pro daný styl zvolíme aktualizování stylu) nebo *načíst styly* z jiného souboru.

V běžném režimu styly používáme tak, že určíme, co má být zformátováno, a pak vybereme styl (buď v rozbalovacím okně na panelu nástrojů stejně jako v MS Wordu, nebo klepneme na styl v okně se seznamem stylů). Když je zapnutý *režim vyplňování formátu*, je jeden styl *aktivní* (zároveň *vybraný* v okně se styly), a pak na cokoli klepneme myši, zformátuje se tímto aktivním stylem.

Styly mohou být *hierarchicky uspořádány*, stejně jako v MS Wordu (viz kap. 1.4.2 na str. 13), i když u předdefinovaných tento postup je použit jen u stylů odstavce. Tato možnost je výhodou především u vytváření vlastních stylů, protože nemusíme určovat všechny vlastnosti nového stylu, některé styl zdědí od svého určeného předchůdce v hierarchii.

3.3 Písmo

3.3.1 Styly znaků

Styly znaků jsou považovány za „čistší“ alternativu ke klepání na tlačítka pro kurzívu, tučné písmo apod., především proto, že když si pro daný typ vyznačení vytvoříme vhodně pojmenovaný styl, máme možnost snadno změnit vlastnosti tohoto stylu pro celý dokument bez nutnosti vyhledávat všechna místa, kde jsme v dokumentu tento typ mohli potřebovat.

Styly znaků používáme jednoduše – vybereme řetězec, který chceme zformátovat daným stylem („vysvítíme“ do bloku) a pak v okně se styly poklepeme na vybraný styl. Pokud je zapnutý režim vyplňování formátu (viz str. 104 v kapitole 3.2), postupujeme přesně naopak – vybereme v seznamu styl a pak klepneme postupně na všechna slova, na která chceme styl použít.

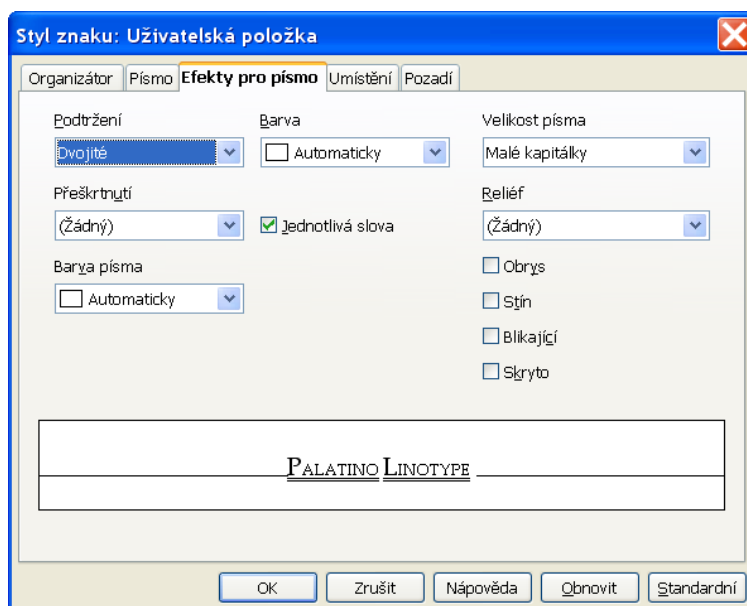
Například v dokumentu o rostlinách můžeme vytvořit styl *název rostliny*, který budeme používat pro názvy rostlin. Ze začátku se rozhodneme psát názvy rostlin kapitálkami, ale po několika stranách nebo desítkách stran nám přijde, že to příliš „bije do očí“, a tak ve stylu změním řez písma na kurzívu nebo nastavíme bezpatkové písmo. Nemusíme procházet celý dokument a hledat názvy rostlin, které jsme do té chvíle napsali, stačí jen na jednom místě změnit vlastnost daného stylu.

V okně Styly a formátování (viz str. 104, obrázek 3.1(b)) zobrazíme seznam stylů pro znaky druhým tlačítkem zleva. Některé styly jsou již předdefinovány, může

se hodit například *Zdůraznění* (standardně kurzíva), *Silné zdůraznění* (standardně tučné písmo), *Internetový odkaz*, *Neproporcionální text* (tzv. písmo psacího stroje).

Existující styl upravíme tak, že v kontextovém menu tohoto stylu zvolíme Upravit. Zobrazí se okno rozčleněné na záložky, na každé záložce upravujeme některou z vlastností. Na první záložce Organizátor je název stylu a jeho předchůdce v hierarchii, od kterého zdědil některé vlastnosti. Při změně předchůdce se přenastaví vlastnosti na dalších záložkách, u předdefinovaných stylů to raději nezkoušíme.

To, co pro písmo můžeme nastavit v MS Wordu, zde najdeme na záložkách Písmo, Efekty pro písmo a Umístění. Najdeme tady prakticky totéž co v MS Wordu, tedy těmito možnostmi se již nebudeme zabývat. Navíc je snad jen možnost zapnutí *kerningu* (podřezávání) na záložce Umístění, vysvětlení pojmu kerning je na straně 7, možnost nastavení rotace (otočení) textu a změna měřítka (třeba zvětšení). Na záložce Pozadí nastavujeme barvu pozadí textu (barva textu samotného se nastavuje na záložce Efekty pro písmo, můžeme zvlášť nastavit i barvu podtržení).



Obrázek 3.2: Nastavení efektů pro písmo ve stylu znaku

Když *vytváříme nový styl*, obvykle nejdříve zvolíme jeho předchůdce v hierarchii. Vybereme takového předchůdce, který se svými vlastnostmi nejvíc blíží našim představám o novém stylu, aby nutných úprav vlastností bylo co nejméně. Tedy v seznamu stylů vybereme předchůdce a v jeho kontextovém menu zvolíme Nový. Zobrazí se dialogové okno, ve kterém určíme vlastnosti nového stylu (především název) a potvrdíme.

Jak si můžeme povšimnout, ve znakových stylech najdeme styl *Zdůraznění*, který je nastaven na kurzívu přesně podle typografických pravidel, proto ho budeme používat pro základní vyznačování. Dále je zde styl *Silné zdůraznění* používaný tehdy, když potřebujeme opravdu hodně upoutat čtenářovu pozornost na dané místo, ten je nastaven na tučné písmo. Také se může hodit styl pro internetovou adresu nebo pro neproporcionální písmo. Běžný text najdeme pod stylem *Výchozí*.

Úlohy:

1. Napište jakýkoliv text a vyzkoušejte na něm různé znakové styly (především *Zdůraznění*, *Silné zdůraznění*, *Neproporcionální text*).
2. Vytvořte znakový styl pro název souboru *Soubor*, nastavte ho na neproporcionální písmo (Courier New), v efektech nastavte *malá písmena*. Vyzkoušejte tento styl v textu.
3. Vytvořte znakový styl pro přípony souborů *Přípona*, nastavte malé kapitálky (karta Efekty pro písmo, položka *Velikost písma*). Opět styl vyzkoušejte v textu.

3.3.2 Zvláštní znaky

Také v OpenOffice.org můžeme využít automatické úpravy formátu při psaní. Najdeme je v menu *Nástroje* → *Automatické opravy*, automatické opravy se povolují v menu *Formát* → *Automatický formát*. Některé speciální znaky najdeme v menu *Vložit* → *Speciální symbol* (nebo využijeme tlačítko na panelu nástrojů).

Jednotlivé znaky se zvláštním významem vložíme následovně (význam pojmů je vysvětlen v kapitole 1.2.2 o zvláštních znacích v MS Wordu na str. 5):

Nezlomitelná meze se vloží klávesovou zkratkou  +  (prostě při psaní mezery stiskneme klávesu ). Tato meze se nerozsířuje.

Pomlčka je vytvářena ze spojovníku automaticky, nastavuje se v Automatických opravách na kartě *Volby*, položka *Nahradit pomlčky*. Dlouhou pomlčku můžeme vyrobit napsáním dvou spojovníků za sebou.

Nedělitelný spojovník (takový, u kterého chceme zakázat pozici na konci řádku) napíšeme klávesovou zkratkou  +  + , například A-Z.

Trojtečka (výpustka) je vytvářena automaticky (opět se nastavuje v Automatických opravách, na kartě *Nahradit*).

Podmíněné dělení slov umožňuje stanovit, kde může být slovo rozděleno, použije se jen při potřebě rozdělení slova (když slovo nemá být rozděleno, dělicí spojovník se nezobrazí). Na tom místě ve slově stiskneme klávesy  + .

Uvozovky jsou ve správném tvaru generovány automaticky, nastavuje se v Automatických opravách, na kartě *Volby*, položka *Nahradit standardní uvozovky vlastními*, případně na kartě *Vlastní uvozovky*. Pro úhlové uvozovky můžeme použít okno *Speciální symbol*.

Slitky a podřezávání (ligatury a kerning) jsou používány automaticky, ale lze je zrušit v dialogu pro nastavení stylu znaků na kartě *Umístění*, je zde zatrhávací políčko *Kerning párů*.

Kapitálky nastavíme ve vlastnostech znaků na kartě *Efekty pro písmo*, položka *Velikost písma*.

3.4 Odstavec

3.4.1 Odrážky a číslování

Na panelu nástrojů máme tlačítka pro odrážky a číslování stejně jako v MS Wordu, můžeme však využít styly pro seznamy, které najdeme opět v okně Styly a formátování. Zde také můžeme vytvářet nové styly podle vlastního uvážení.

Styl seznamu můžeme použít i na již existující položky tak, že v textu vybereme řádky s budoucím obsahem seznamu a v okně Styly a formátování poklepeme na vybraný styl. Každý řádek se zobrazí ve vlastní položce. Odsazení měníme buď tlačítky na panelu nástrojů nebo klávesami  a  (máme kurzor nastavený na začátku položky).

V předdefinovaných stylech si moc nevybereme, zejména když chceme použít víceúrovňové číslování pro nadpisy.

Příklad 3.1 Vytvoříme nový styl pro víceúrovňové číslování, který můžeme použít pro číslování nadpisů (ve skutečnosti se však číslování nadpisů provádí trochu jinak, jak uvidíme v kapitole 3.4.2).

1. V okně Styly a formátování klepneme na ikonu *Styly seznamu*.
2. Vyvoláme kontextové menu (stačí klepnout kamkoliv vedle zobrazených stylů) a vybereme *Nový*.
3. Hned na první kartě styl nazveme *Víceúrovňové*.
4. Přesuneme se na kartu *Osnova* a vybereme styl nejvíce se blížící našim potřebám (arabské číslice, mezi čísly úrovní tečka, za posledním číslem není tečka).
5. Přesuneme se na kartu *Umístění* a u prvních tří úrovní nastavíme vlastnost *Odsazení* na 0cm a vlastnost *Minimální mezera mezi číslováním a textem* na 0,4cm.

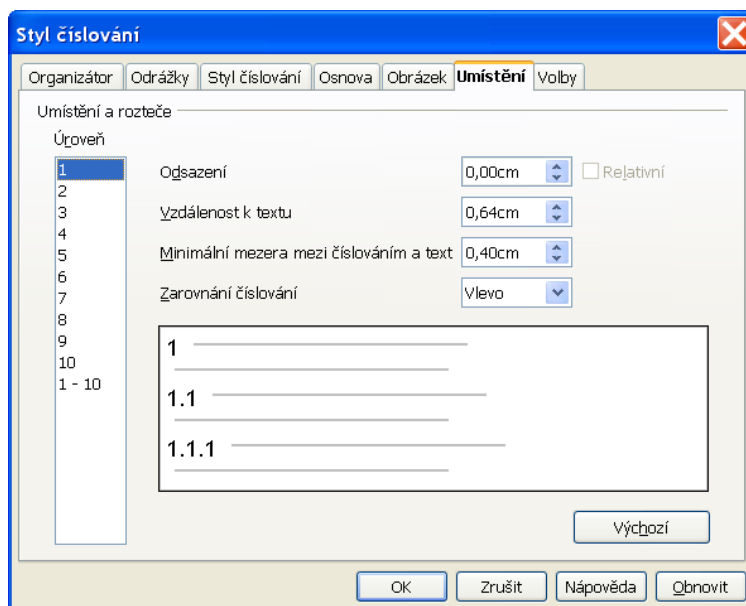
To by už stačilo, ale ještě si vyzkoušíme jednoduché „vyšperkování“ číslování kapitol první úrovně – chceme, aby před číslem bylo slovo „Kapitola“. To zajistíme na kartě *Volby*, kde nejdříve zvolíme úroveň 1 a pak v poli *Před* zadáme *Kapitola*.

Ted' už můžeme styl vyzkoušet.

3.4.2 Styly odstavce

Ve stylech odstavce najdeme poměrně dost základních použitelných stylů, ale pro naše účely bude nutné některé upravit.

Zde se uplatňuje dědění v hierarchii více než v jiných stylech. Pro běžný text zde jako základní styl najdeme *Výchozí*, jeho potomkem je styl *Tělo textu*, potomkem tohoto stylu je zase *Odsazení prvního řádku*, *Odsazení těla textu*, *Poznámka na okraj* a další. Potomkem stylu *Výchozí* je i styl pro nadpisy. Proto pokud budeme chtít měnit vlastnosti (třeba typ písma) všech stylů odstavců pro běžný text a seznamy,



Obrázek 3.3: Nastavení víceúrovňového číslování

provedeme změnu u stylu *Tělo textu*, a když je nutné přibrat i nadpisy, změníme styl *Výchozí*.

Když tedy budeme chtít používat pro běžný text odstavce s odsazením prvního řádku, využijeme styl *Odsazení prvního řádku*, dále pokud se rozhodneme první odstavec každé kapitoly na prvním řádku neodsazovat, vytvoříme potomka buď stylu *Tělo textu* nebo stylu *Odsazení prvního řádku* a nastavíme odsazení na 0.

Podobně můžeme pracovat i s nadpisy. Nadpisy všech úrovní jsou potomky stylu *Nadpisy*, proto když chceme měnit vlastnosti zároveň styly pro všechny úrovně nadpisů, změníme styl *Nadpisy*.

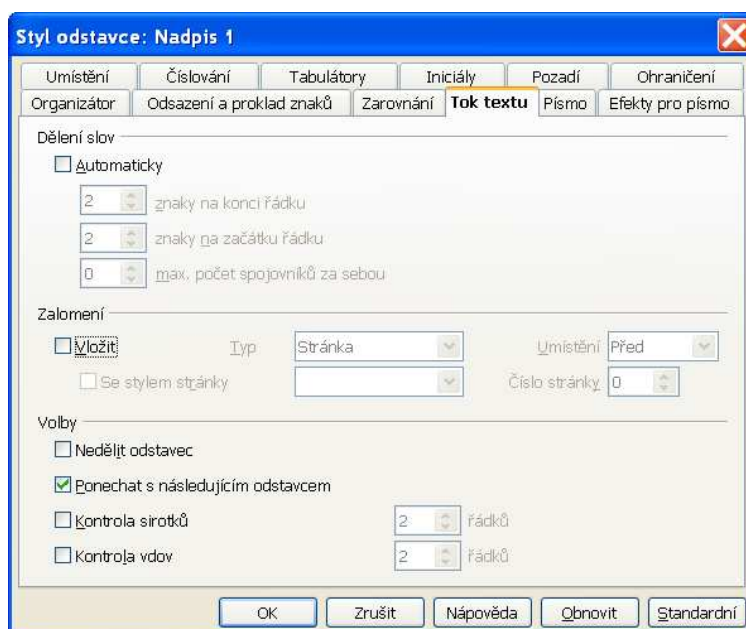
K vlastnostem stylů pro odstavce se dostaneme stejně jako u předchozích probraných. Zobrazíme okno *Styly a formátování* (třeba klávesou **F11**) a nahoře klepneme na první tlačítko. Pak stačí u upravovaného stylu vyvolat jeho kontextové menu a zvolit *Upravit*.

Počet záložek je mnohem vyšší než u stylů pro znaky a seznamy, můžeme toho tedy nastavovat mnohem více. Základní informace jsou opět na záložce *Organizátor*, tam najdeme kromě názvu stylu také styl, kterým se má pokračovat po použití tohoto stylu v textu a stisknutí klávesy **Enter** (*Další styl*) a jeho předchůdce v hierarchii (*Spojeno s*), dále vedle názvu je zaškrtačací políčko *Automatická aktualizace*, které, když je zaškrtnuto, způsobí, že změny na odstavcích v dokumentu se projeví i na definici stylu.

Na kartě *Odsazení a proklad znaků* se nastavuje odsazení odstavce od levého a pravého okraje, odsazení prvního řádku, vertikální odsazení od okolních odstavců a řádkování. Dále zde najdeme volbu *Registr*. Když je nastaven registr, všechny řádky jsou stejně vysoké, tedy pokud vložíme znak výrazně vyšší než ostatní znaky, řádek zůstane pořád stejně vysoký. Aby tato volba fungovala, musí být tato vlastnost nastavena i u stylů stránky (tam se podíváme později).

Karta Zarovnání pro nás taky nebude moc velkým překvapením, najdeme tam nastavení zarovnání odstavce vzhledem k okrajům stránky. Zajímavá je volba aktivovaná při nastavení zarovnání do bloku, Rozpólit jedno slovo. Když určíme, že poslední řádek odstavce se má taky zarovnat do bloku a zaškrtneme tuto volbu, v případě, že na posledním řádku zůstane jen jediné slovo, jsou jeho znaky „rozstrkány“ po celé délce odstavce, tedy na znaky se použije to, co by se jinak použilo na slova.

Na kartě Tok textu (je na obr. 3.4 na str. 109) nastavujeme chování konce řádků, konce odstavce a odstavce na konci strany. V první části záložky můžeme zajistit, že na konci a případně na začátku řádku nebudou „osamělá písmenka“ včetně jednopísmenných předložek a spojek, a taky že v případě dělení slov nebude na koncích řádků přímo nad sebou více spojovníků (z estetických důvodů).



Obrázek 3.4: Nastavení toku textu odstavce

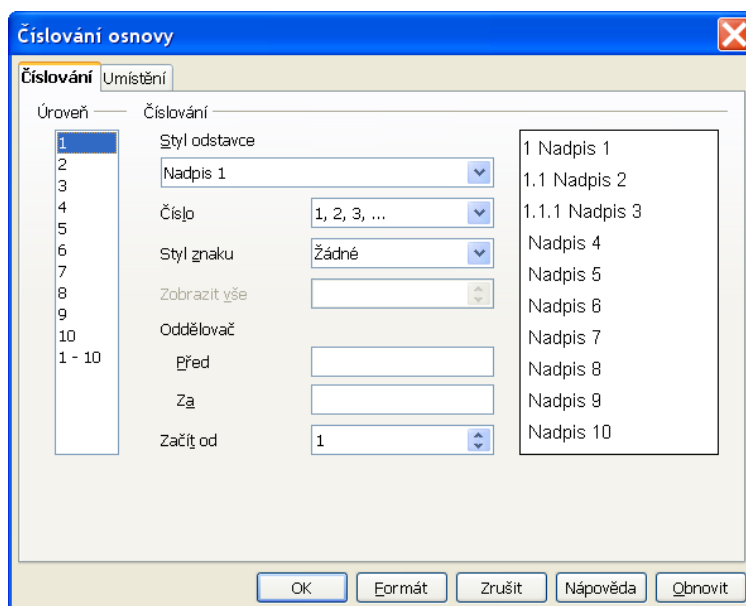
V další části této karty určujeme, co se má stát, když ukončíme odstavec (stiskneme klávesu **Enter**). Můžeme zde ošetřit práci se sloupci, ale používanější volbou je *Stránka*. Když vložíme konec stránky před tento odstavec (třeba pro styl nadpisu první úrovně), zajistíme, že odstavec bude vždy začínat na nové straně. Navíc zde můžeme určit styl následující strany včetně čísla strany (když je číslo nastaveno na 0, pokračuje se v číslování stran od předchozí strany). Zde tedy nastavujeme to, pro co bychom museli v MS Wordu vložit nový oddíl.

V třetí části karty se nastavuje chování odstavce na konci strany. Je to totéž co nastavujeme v MS Wordu na kartě Tok textu ve vlastnostech odstavce, tedy možnost odstavce vůbec nedělit koncem strany (celý se přesune na následující stranu), ponechat s následujícím odstavcem (například pro nadpisy, aby za nadpisem na každé straně následoval začátek alespoň jednoho odstavce), nedovolit vznik *sirotek* (osamělý řádek na konci strany) a *vdov* (osamělý řádek na začátku strany, většina odstavce je na předchozí straně).

Karty Písmo, Efekty pro písmo a Umístění jsou stejné jako u stylů znaků.

Karta *Číslování* se nám může hodit především při zajišťování číslování řádků, případně když chceme využít trochu složitější cestu pro nastavení číslování nadpisů, můžeme využít styl vytvořený v příkladu 3.1 na str. 107, zde (postupně pro styly používaných úrovní nadpisů) pak v rozbalovacím seznamu zvolíme tento styl a tak napojíme nadpis dané úrovně na číslování.

Jednodušší způsob zajištění číslování nadpisů nespočívá v nastavení stylů, ale stačí v menu programu najít *Nástroje* → *Číslování osnovy*. V zobrazeném okně postupně vybíráme úroveň, které budeme používat, a v rozbalovacím seznamu *Číslo* zvolíme arabské číslice (1, 2, ...).



Obrázek 3.5: Nastavení číslování nadpisů

Příklad 3.2 Upravíme styly odstavce následovně:

1. Nastavíme pro běžný text (tj. ve stylu *Tělo textu*) typ písma na Book Antiqua a zarovnání do bloku. Tato změna se projeví také u všech potomků stylu *Tělo textu*, včetně stylu *Odsazení prvního řádku*.
2. Klepneme pravým tlačítkem myši na styl *Tělo textu* a vybereme v kontextovém menu volbu *Nový*. Zobrazí se okno pro vytvoření nového stylu, který je potomkem stylu *Tělo textu*. Styl pojmenujeme *První odstavec* a určíme, aby za tímto odstavcem automaticky následoval odstavec se stylem *Odsazení prvního řádku* (na kartě *Organizátor* nastavíme volbu *Další styl*). Pak potvrdíme.
3. Pozměníme styl pro nadpisy. Stejně jako u předchozích úprav, v kontextovém menu stylu *Nadpis* vybereme *Upravit* a na kartě *Organizátor* nastavíme *Další styl* na *První odstavec*. Tuto vlastnost zdědí nadpisy všech úrovní.

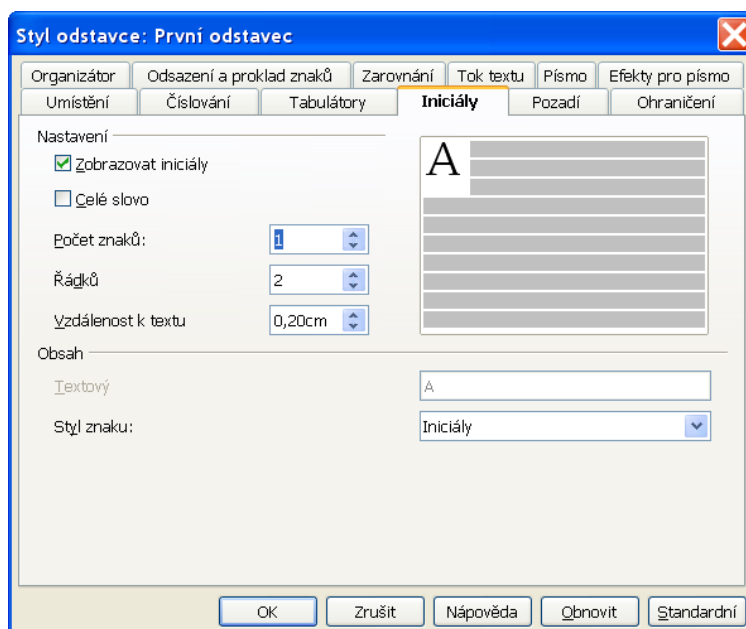
4. Zajistíme číslování nadpisů. V menu aplikace najdeme Nástroje → Číslování osnovy a v zobrazeném okně nastavíme pro několik prvních úrovní nadpisů arabské číslování (volba *Číslo*).
5. Všechny nastavené styly vyzkoušíme – do dokumentu vložíme nadpisy několika úrovní, za každý začneme psát text běžného odstavce. První odstavec každé kapitoly by neměl mít odsazený první řádek, ostatní odstavce naopak ano.

3.4.3 Iniciály

Používání iniciál je zajímavá možnost, jak „vyšperkovat“ dokument. Můžeme určit, že první písmeno, několik písmen nebo třeba celé první slovo bude zformátováno jinak než následující text. Tato vlastnost se nastavuje ve vlastnostech odstavcových stylů na kartě Iniciály, ovšem použijeme ji spíše jen pro první odstavec každé kapitoly (nebo první odstavec hlavní kapitoly).

Postupujeme tak, že nejdřív si vytvoříme *znakový styl* pro naši iniciálu (můžeme jen upravit existující styl *Iniciály*), třeba změníme řez písma, nastavíme stín apod. (nezvětšujeme, to se provede automaticky při použití v odstavci), a pak v příslušném odstavcovém stylu na kartě Iniciály nastavíme vše potřebné včetně vazby na náš znakový styl.

Příklad 3.3 Budeme pokračovat v příkladu 3.2 z předchozí kapitoly, styl *První odstavec* obohatíme o iniciálu.

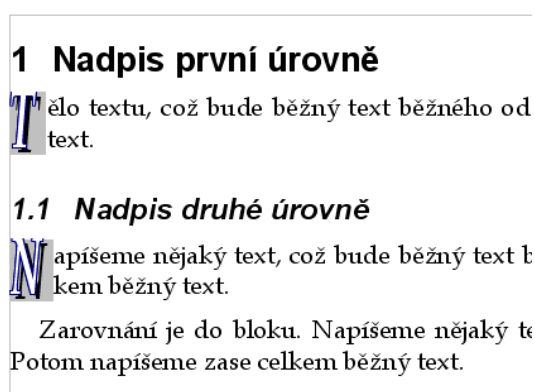


Obrázek 3.6: Nastavení iniciály pro odstavec

6. V znakových stylech najdeme styl *Iniciály*. Zvolíme v jeho kontextovém menu Upravit a přejdeme na kartu Písmo. Tam nastavíme typ písma na Book Antiqua

a tučnou kurzívu. Na kartě Efekty pro písmo určíme modrou barvu písma a dále zaškrtneme volby *Obrys* a *Stín*. Potvrdíme.

7. Ve stylech odstavce najdeme styl *První odstavec*. V jeho kontextovém menu taktéž zvolíme Upravit a přejdeme na kartu Iniciály. Tam zaškrtneme volbu *Zobrazovat iniciály*, nastavíme volbu *Řádků* na 2 řádky, *Vzdálenost k textu* na 0,20 cm a pro *Styl znaku* vybereme v seznamu *Iniciály*. Karta by měla vypadat jako na obrázku 3.6. Potvrdíme.
8. Vyzkoušíme. Výsledek by měl vypadat tak jako na obrázku 3.7. Při tisku se samozřejmě šedý obdélník pod iniciálou neobjeví, to si můžeme ověřit v náhledu pro tisk (v menu Soubor → Náhled strany).



Obrázek 3.7: Ukázka použití iniciály pro první odstavec

3.5 Struktura dokumentu

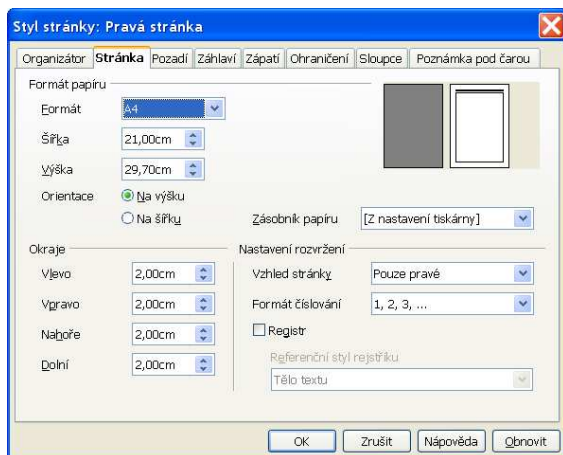
3.5.1 Styly stránky

Taktéž pro stránky máme styly. Pracujeme s nimi jako s jinými styly, tedy v okně Styly a formátování máme k dispozici seznam stylů (styly stránky najdeme pomocí čtvrtého tlačítka zleva na horním okraji tohoto okna) a možnosti jejich úprav či vytvoření nového stylu. Dále na spodním okraji hlavního okna aplikace, tedy na *stavovém panelu*, máme možnost vybírat si mezi standardními styly stran.

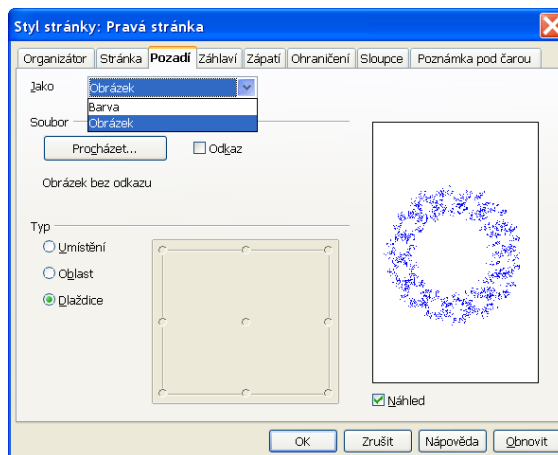
Podíváme se na možnosti nastavení stylů stránky. Po zobrazení okna s vlastnostmi stylu (můžeme vyzkoušet třeba pro styl *Pravá stránka*) na kartě Organizátor najdeme kromě názvu stylu také informaci o případném předkovi (*Spojeno s*), ale především nás zajímá položka *Další styl*. Zde můžeme zajistit například střídání stylů pro levou a pravou stranu jednoduše tak, že ve stylu pro pravou stranu zde vybereme styl pro levou a naopak.

Na kartě Stránka lze změnit rozměry strany a jejích okrajů, dále můžeme zvolit mezi orientací na výšku nebo na šířku a taky zvolit *zrcadlení* stránek. Zrcadlení použijeme, když potřebujeme například číslo stránky vždy u vnějšího okraje. Na této kartě se také nastavuje formát číslování stránek (jen formát, číslování samotné se zapíná a nastavuje jinak).

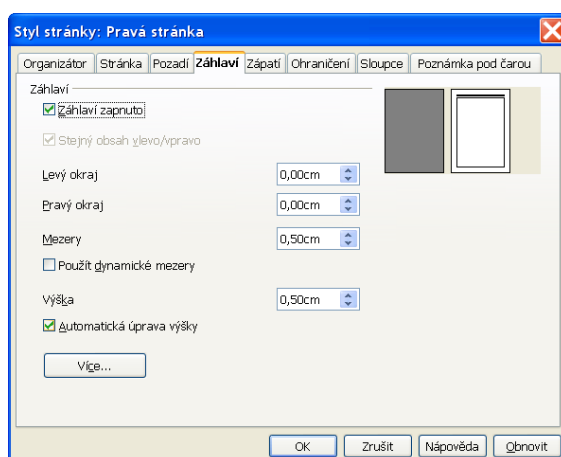
Karta Pozadí slouží ke stanovení barvy pozadí, a také zde můžeme umístit obrázek na pozadí, který může fungovat také jako jakýsi vodoznak.



(a) Základní vlastnosti stránky



(b) Obrázek na pozadí stránky



(c) Zapnutí záhlaví stránky

Obrázek 3.8: Nastavení stylu stránky

Karty Záhlaví a Zápětí slouží k nastavení vzhledu záhlaví a zápatí stránky. Nejdříve musí být aktivováno jejich zobrazení, to uděláme zatržením pole hned na začátku příslušné stránky. Pak nás čekají celkem obvyklé volby, tedy rozměry okrajů (můžou tedy být jiné než u zbytku stránky), případně mezer. Přes tlačítko Více můžeme určovat ohraničení a stínování záhlaví a také barvu pozadí.

Na kartě Ohraničení nastavujeme ohraničení a stínování celé stránky, karta Sloupce slouží ke stanovení počtu a formátu sloupců na stránce (včetně případné dělicí čáry) a na kartě Poznámky pod čarou se určuje formát poznámek pod čarou.

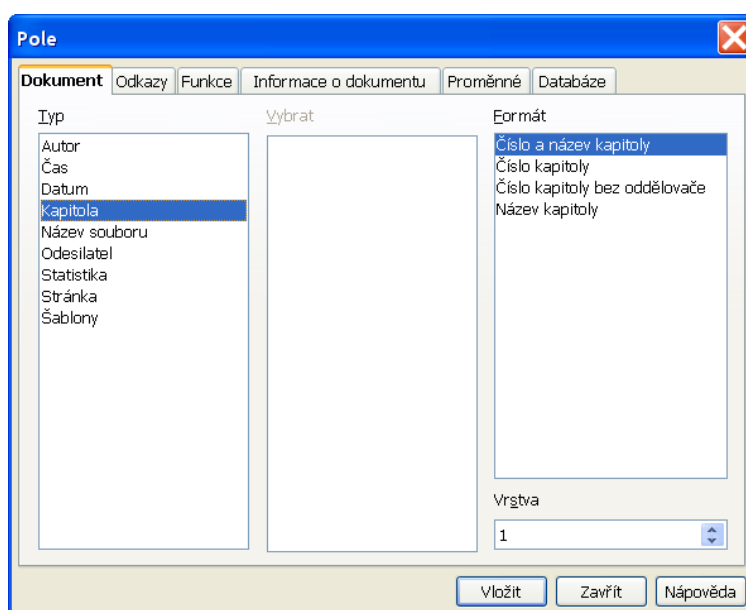
3.5.2 Záhlaví a zápatí

Záhlaví a zápatí můžeme nastavovat, když ve vlastnostech daného stylu stránky na kartě Záhlaví resp. Zápětí zaškrtneme pole *Záhlaví zapnuto* resp. *Zápatí zapnuto*. Po

zapnutí, případném nastavení dalších vlastností a potvrzení okna se v dokumentu objeví rámečky, do kterých můžeme buď psát přímo text nebo vkládat obrázky, a nebo můžeme vkládat pole.

Pro záhlaví a zápatí existují zvláštní styly odstavce. Proto pokud chceme mít třeba v záhlaví text vždy tučným písmem určité velikosti nebo třeba kapitálkami, stačí jen pozměnit vlastnosti stylu *Záhlaví*.

U polí máme možnosti obdobné možnostem v MS Wordu. Lze vložit číslo stránky, počet stran, čas, datum, jméno autora, název dokumentu, atd. Ke vkládání polí se dostaneme v menu Vložit → Pole, kde vybereme buď přímo určitý typ pole, nebo volbou Jiné přejdeme do pokročilejšího rozhraní se spoustou různých typů polí, kde nás zřejmě bude nejvíc zajímat záložka Dokument.



Obrázek 3.9: Nástroj pro pokročilejší vkládání polí

Můžeme si všimnout, že v nabídce polí je i *Kapitola*. Takto vložíme do záhlaví třeba název hlavní kapitoly, ovšem stejně jako v MS Wordu jde vlastně o křížový odkaz na konkrétní kapitolu. Je tedy zajištěno aktualizování v případě přejmenování této kapitoly, ale není zajištěna automatizace obecného nastavení na hlavní kapitolu. Proto v tom případě musíme vytvořit zvlášť styly stran pro různé hlavní kapitoly (to není problém, stačí vytvořit „abstraktní“ styl s nastavením všech vlastností a pak vytvářet potomky lišící se jen v tomto poli).

3.5.3 Přechody mezi stránkami

V kapitole 3.5.1 jsme si vysvětlili jeden ze způsobů, jak určit, jaký typ stránky bude následovat, stačilo ve vlastnostech stylu nastavit *Další styl* na příslušnou hodnotu. Podobnou možnost nám také nabídly styly odstavce, kde na kartě Tok textu jsme mohli při nastavení vložení stránky před odstavce určit styl vkládané stránky (třeba styl stránky, na které mají začínat hlavní kapitoly).

V menu Vložit → Ruční zalomení lze vložit konec řádku, sloupce nebo stránky, u konce stránky můžeme určit styl, který má následovat.

Obdobou oddílů z MS Wordu jsou zde *sekce*. Pro nastavení vlastností sekce máme však k dispozici mnohem více voleb než v případě oddílů MS Wordu. Sekce může být například zamknutá, pouze pro čtení, napojená na vnější objekt (vazba DDE), nastavujeme sloupce, odsazení odstavce (třeba pro citáty), užitečná může být také možnost určit umístění poznámek pod čarou a vysvětlivek.

Novou sekci vložíme z menu Vložit → Sekci. V zobrazeném okně určíme všechny potřebné vlastnosti a potvrdíme. Poznámky pod čarou můžeme umísťovat buď na konec sekce nebo na konec celé stránky, vysvětlivky opět buď na konec sekce nebo na konec dokumentu. Číslování obojího může být nezávislé na zbytku dokumentu. Pokud zvolíme u poznámky pod čarou umístění na konec stránky (tj. necháme výchozí volbu), sekce se prodlouží až ke konci stránky.

3.6 Odkazy a popisky objektů

3.6.1 Křížové odkazy

3.6.2 Tabulky

3.6.3 Obrázky

3.6.4 Popisky objektů a odkazy

3.7 Dokumentové seznamy

3.8 Šablony

KAPITOLA 4

Co ještě přidám

- Další programy kancelářských balíků (Excel, Calc, PowerPoint, Impress, atd.), velké DTP programy (Scribus, Adobe InDesign, QuarkXPress, PageMaker, . . .)
- Jak získat legální software zdarma (licence, OpenSource, free software, free-ware, . . .)
- Jak na informace (získávání, ověřování, zpracování, zveřejňování informací)

Přílohy

PŘÍLOHA A

Užitečné klávesové zkratky

PŘÍLOHA B

Další informace o L^AT_EXu

B.1 Nejdůležitější příkazy L^AT_EXu

B.2 Délkové registry a čítače v T_EXu

Čísla, která se často mění a je nutno je evidovat a odkazovat se na ně, se ukládají v čítačích. Čítače si můžeme představit jako jakési zástupné názvy pro tato čísla, přes které můžeme k číslům přistupovat bez nutnosti znalosti jejich momentální hodnoty.

Číslo stránky je uloženo v čítači `\thepage`. Tento čítač je tedy nulován při každém použití příkazu `\pagenumbering`, ale můžeme s ním pracovat i jinak, stejně jako s jinými čítači.

B.3 Zajímavé třídy a balíčky

Závěr

LITERATURA

— Citace a typografie —

- [C1] BOLDIŠ, P.: *Bibliografické citace dokumentů podle ČSN ISO 690 a ČSN ISO 690-2: Část 1, Citace: metodika a obecná pravidla*. Verze 3.3 [online].
URL: <http://www.boldis.cz/citace/citace1.ps>,
<http://www.boldis.cz/citace/citace1.pdf>
[cit. 20. 10. 2006]
- [C2] BOLDIŠ, P.: *Bibliografické citace dokumentů podle ČSN ISO 690 a ČSN ISO 690-2: Část 2, Modely a příklady citací u jednotlivých typů dokumentů*. Verze 3.0 [online].
URL: <http://www.boldis.cz/citace/citace2.ps>,
<http://www.boldis.cz/citace/citace2.pdf>
[cit. 20. 10. 2006]
- [C3] JANÁK, R.: *Typografie*. [online], 2001.
URL: <http://www.typografie.unas.cz/>
[cit. 4. 12. 2006]
- [C4] SÁRKÖZI, R.: *Jak se dělá školní časopis (3) – Počítačová typografie*. Česká škola [online], 2004.
URL: <http://www.ceskaskola.cz/Ceskaskola/AR.asp?ARI=3212&CAI=2125>
[cit. 5. 2. 2007]
- [C5] OLŠÁK, P.: *Kam se poděla dobrá typografie?* In: *Mensa* 5/96, s. 37–42. [online], 1996.
URL: <http://petr.olsak.net/ftp/olsak/typo/mensa.pdf>
[cit. 5. 2. 2007]

— Diplomové práce —

- [D6] NOVÁK, J.: *Doporučení pro úpravu závěrečných prací*. Diplomová práce. Brno, Mendelova zemědělská a lesnická univerzita [online], 2006.
URL: <http://old.mendelu.cz/~rybicka/zpract/jaknadip.pdf>
[cit. 25. 10. 2006]

- [D7] POLÁCH, E.: *Pravidla sazby diplomových prací*. České Budějovice, Pedagogická fakulta Jihočeské univerzity [online], 1998.
<http://home.pf.jcu.cz/~edpo/pravidla/pravidla.pdf>
[cit. 25. 10. 2006]
- [D8] *Studentův wikipřůvodce životem na ČVUT – Diplomová práce*. Praha [online], 2006.
URL: <http://www.student.cvut.cz/cwut/>, do pole Hledat v levé části okna zadejte text Diplomová práce
[cit. 25. 10. 2006]
- [D9] SYSEL, P.: *Šablona pro diplomové práce*. Fakulta elektrotechniky a komunikačních technologií, VUT Brno [online].
URL:
<http://www.utko.feec.vutbr.cz/latex/index.php?page=download#template>
[cit. 8. 11. 2006]
- [D10] Vysoká škola báňská-Technická univerzita, Fakulta elektrotechniky a informatiky. *Závazné pokyny pro vypracování bakalářské a diplomové práce*. Ostrava [online], 2005.
URL: http://fei.vsb.cz/www/dokumenty/szz_dp.html
[cit. 25. 10. 2006]

— Programy, instalace a návody —

- [P11] České stránky firmy Adobe [online].
URL: <http://www.adobe.cz>
[cit. 2. 11. 2006]
- [P12] Nakladatelství Computer Press [online].
URL: <http://knihy.cress.cz>
[cit. 28. 10. 2006]
- [P13] Slunečnice [online].
URL: <http://www.slunecnice.cz>
[cit. 2. 11. 2006]
- [P14] VAVREČKOVÁ, Š.: *Instalace a počeštění systému Mik $\text{\TeX}$ verze 2.5 ve Windows*. Opava [online], 2006.
URL: <http://fpf.slu.cz/~vav10ui/obsahy/dipl/instalace25.pdf>
[cit. 8. 11. 2006]
- [P15] *LyX for Windows*. [online].
URL: <http://wiki.lyx.org/Windows/Windows>
[cit. 29. 11. 2006]
- [P16] MATOUCH, P.: *Instalace $\text{\TeX}$ u pro Windows*. [online].
URL: <http://dce.felk.cvut.cz/roubal/teaching/LaTeX/TeXinst.pdf>
[cit. 1. 12. 2006]

- [P17] *Specifika vývoje českých dokumentů v LaTeXu, konfigurace vývojového prostředí a jeho individuální přizpůsobení potřebám a zvyklostem* [online].
URL: <http://www.fi.muni.cz/~sojka/PB029/2006cv2.html>
[cit. 1. 12. 2006]
- [P18] Nakladatelství Springer: *Lecture Notes in Computer Science*. [online].
URL: <http://www.springer.com/east/home/computer/lncs?SGWID=5-164-7-72376-0>
[cit. 5. 12. 2006]
- [P19] The CTAN team: *CTAN Search*. Searching $\text{T}_{\text{E}}\text{X}$ and $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ packages. [online].
URL: <http://tug.ctan.org/find.html>
[cit. 11. 1. 2007]
- [P20] *Inkscape Homepage*. [online]
URL: <http://www.inkscape.org/>
[cit. 5. 2. 2007]
- [P21] *GNUPlot Homepage*. [online], 2006.
URL: <http://www.gnuplot.info/>
[cit. 5. 2. 2007]
- [P22] PINKAS, P.: *GNUPlot: seznamte se*. Článek na Root.cz, [online], 2001.
URL: <http://www.root.cz/clanky/gnuplot-seznamte-se/>
[cit. 5. 2. 2007]
- [P23] *L^AT_EX_PiX Homepage*. [online]
URL: <http://members.home.nl/nickvanbeurden/latexpik.htm>
[cit. 5. 2. 2007]
- [P24] *T_EXCad32 Homepage*. [online]
URL: http://www.das-gelbe-rechenbuch.de/Textcad32/Index_e.html
[cit. 5. 2. 2007]
- [P25] *T_EXCAD Homepage*. [online]
URL: <http://homepage.sunrise.ch/mysunrise/gdm/textcad.htm>
[cit. 5. 2. 2007]

— MS Word —

- [W26] KUBÁLEK, T., TOPOLOVÁ, I.: *Manažerská informatika. Textový procesor Microsoft Word verze 2000 CZ*. Praha, VŠE [online], 2001.
URL: http://fph.vse.cz/fakulta/informatizace/skripta_MSWord2000.asp
[cit. 20. 10. 2006]
- [W27] RYBIČKA, J.: *Zpracování textů počítačem*. [online]
URL: old.mendelu.cz/~rybicka/zpract/mikrotyp.doc
[cit. 5. 2. 2007]
- [W28] OLŠÁK, P.: *T_EX kontra Word*. [online]
URL: <http://petr.olsak.net/ftp/olsak/typo/textword.pdf>
[cit. 5. 2. 2007]

- [W29] VOJÁČEK, K.: *Zpracování textů textovým procesorem MS Word*. KAVOJ Publishing, [online], 2006.
URL: http://www.mgplzen.cz/download/ivt_word.pdf
[cit. 5. 2. 2007]

— $\text{\TeX}$ a $\text{\LaTeX}$ —

- [T30] KROB, J.: *Stránky o $\text{\TeX}$* [online].
URL: <http://www.phil.muni.cz/~jokr/tex.html>
[cit. 6. 11. 2006]
- [T31] RYBIČKA, J. *$\text{\LaTeX}$ pro začátečníky*. Brno, Konvoj, 2003.
- [T32] ŠVEJDAR, V.: *O bibliografickém systému Bib $\text{\TeX}$* . UK Praha [online], 2005.
URL: <http://www1.cuni.cz/~svejdar/texdev/BibTeXInfo.pdf>
[cit. 8. 11. 2006]
- [T33] PAKIN, S.: *The Comprehensive $\text{\LaTeX}$ Symbol List*. [online], 2005.
URL: [odkaz](#)
(<http://www.ctan.org/tex-archive/info/symbols/comprehensive/symbols-a4.pdf>) [cit. 29. 11. 2006]
- [T34] HARDERS, H.: *Symbols in Textcomp*. [online], 2005.
URL: <http://www.tug.org/tex-archive/info/symbols/text/textcomptst.ps>
[cit. 29. 11. 2006]
- [T35] Československé sdružení uživatelů $\text{\TeX}$. [online].
URL: <http://www.cstug.cz/>
[cit. 29. 11. 2006]
- [T36] BEDNÁŘ, R.: *$\text{\LaTeX}$ manuál*. [online].
URL: <http://www.cstug.cz/latex/lm/frames.html>
[cit. 29. 11. 2006]
- [T37] SADOVSKÝ, P. a kol.: *Jak nejrychleji napsat svůj první dokument v $\text{\LaTeX}$* . [online].
URL: <http://fyzika.ufyz.feec.vutbr.cz/~petrsad/download/LaTeX/JakPsat.pdf>
[cit. 29. 11. 2006]
- [T38] OETIKER, T., PARTL, H., HYNÁ, I., SCHLEGL, E., KOČER, M., SÝKORA, P.: *Nepřiliš stručný úvod do systému $\text{\LaTeX} 2_{\epsilon}$* . [online].
URL: http://fyzika.ufyz.feec.vutbr.cz/~petrsad/LaTeX/lshort_czech.pdf
[cit. 29. 11. 2006]
- [T39] SPOUCH, P., FORMÁNEK, P.: *$\text{\LaTeX}$ v kostce*. [online].
URL: <http://www.it.cas.cz/manual/latex/>
[cit. 29. 11. 2006]

- [T40] KOVÁŘ, P.: *Úvod do T_EXu (pro každého něco)*. [online].
URL: <http://homel.vsb.cz/~kov16/talks/tex/>
[cit. 29. 11. 2006]
- [T41] ROBERTS, A.: *Getting to Grips with L^AT_EX*. [online].
URL: <http://www.andy-roberts.net/misc/latex/index.html>
[cit. 29. 11. 2006]
- [T42] DOOB, M.: *A Gentle Introduction to T_EX*. A Manual for Self-study. Department of Mathematics, The University of Manitoba, Winnipeg, Manitoba, Canada. [online].
URL: <ftp://ftp.cstug.cz/pub/tex/CTAN/info/gentle/gentle.pdf>
[cit. 10. 1. 2007]
- [T43] OSWALD, U.: *L^AT_EX Graphics*. Petersburg. [online].
URL:
<http://www.ursoswald.ch/LaTeXGraphics/overview/overview.html>
[cit. 10. 1. 2007]
- [T44] Indian T_EX Users Group: *Online Tutorials in L^AT_EX*. [online].
URL: <http://www.tug.org.in/tutorials.html>
[cit. 10. 1. 2007]

— OpenOffice.org —

- [O45] OpenOffice.org Homepage. [online]
URL: <http://www.openoffice.org/>
[cit. 7. 2. 2007]
- [O46] Domovská stránka OpenOffice.org v češtině. [online]
URL: <http://www.openoffice.cz/>
[cit. 7. 2. 2007]
- [O47] Open Document. Čánek na wikipedia.org, [online], 2006.
URL: <http://cs.wikipedia.org/wiki/OpenDocument>
[cit. 7. 2. 2007]

REJSTŘÍK POJMŮ

A

abstract 19
abstrakt 19, 39, 61
aktualizace
 automatická 16
 dokumentového seznamu 26
 odkazu 26
 stylů ze šablony 13
autor 61

B

bad box 70
balíček 40, 48, 50, 57, 59, 71
barva 85
 kreslení 85
 nová 85
 pozadí 85
 pozadí stránky 85
 předdefinovaná 85
 textu 85
 vlastní 86
beletrie 56
bibliografie 18
blok 66
bod typografický 5, 44, 74
box 77, 84, 85
 vadný 70

C

chyba 70
citace 10, 29
čára
 plná 49

 pružná 49, 79
 tečkovaná 49, 79
část (part) 20, 60
číslování
 nadpisů 13, 59
 stránek 19, 62
 víceúrovňové 13, 55
 víceúrovňové prostředí 83
čítač 69
článek 56
článek odborný 19, 56
členění
 na části 20, 60
čtverčík 10, 44

D

databáze balíčků 59
datum 61
diakritika 51
distribuce T_EXu 40, 42, 56
dodatek 18
dokument
 hlavní 33, 65, 66
 jednostranný .. 17, 22, 56, 59, 62, 63
 oboustranný 17, 22, 56, 59, 62, 63
 struktura 17, 65
 vnořený 33
dokumentace k balíčkům 59
dopis 56
doslov 18

E

e_{TeX}.....41
 editor grafický73
 error.....70

F

font 4, 47, 57
 formát souboru35
 cls.....59
 DOC.....35
 DOT 12, 13, 35
 DVI.....39, 43, 65, 85, 87
 EPS.....71, 73
 idx 89
 JPG 71
 lof.....89
 LOG.....70
 lot 89
 PDF 35–37, 39, 43, 65, 71–73, 85,
 87
 PNG.....71
 PPT.....35
 PS.....39, 43, 65, 71–73, 85, 87
 RTF 35
 sty.....59
 TCP.....66
 TEX 39, 42, 65, 73
 toc.....89
 TPS.....66
 TXT 35
 uzavřený.....34
 XLS 35

H

hierarchie stylů.....13
 hlášení varovné.....69
 hlavička dokumentu42

I

ikonka.....8, 62
 vyznačující.....8
 instalace
 balíčku59
 odložená.....3
 ISBN 17
 italika 7

K

kapitálka 7, 8, 13, 30, 46
 kerning.....7, 50
 kniha odborná.....56
 komentář 48, 50
 kurzíva 7, 13, 30, 46
 kuželka 5, 10, 44, 52
 křivka
 Bezierova 76

L

La_{TeX} 39, 40
 leták.....56
 ligatura 7, 50
 literatura 18, 29

M

manuál referenční.....56
 mezer
 dlouhá 32
 dlouhá horizontální 49
 mezi odstavci 10
 nezlomitelná 6, 49, 51, 55
 pevná 6, 49
 pružná 49
 vertikální 49, 55
 zúžená.....6
 mezititul 21
 Mik_{TeX}.....41, 42, 59
 minuska.....7
 místo v dokumentu 69

N

nadpis
 číslovaný 11, 13, 60
 nečíslovaný.....60, 89
 v obsahu.....60
 typ písma.....5, 12
 název dokumentu 61

O

obdélník vybarvený 49
 objekt
 otočení.....72
 roztáhnutí 72
 obrázek 82
 vkládání 82
 vložení.....71, 72

obsah 18, 27, 60, 61, 65, 89
 obtékání objektů 83
 oddíl 22
 odkaz 23, 29
 do seznamu literatury 87, 88
 hypertextový 86
 křížový 23, 69, 71
 na číslo stránky 23, 69
 na kapitolu 24, 69, 87
 na obrázek 25, 69
 na položku seznamu 24, 69
 na prostředí označené číslem 70
 na stranu 87
 na tabulku 25, 69
 v obsahu hypertextový 87
 odsazení
 odstavce vertikální 10, 16, 30, 52
 prvního řádku odstavce 10, 52
 odstavec pojmenovaný 53
 orámování 77
 objektů 84
 osnova 3, 34

P

patitul 17
 patka 5
 písmo
 bezpatkové 5, 8, 46
 groteskové 5
 kaligrafické 5
 neproporcionální 5, 8, 46, 86
 patkové 5, 46
 polotučné 46
 proporcionální 5
 serifové 5
 skloněné 7, 46
 tučné 7, 46
 vzpřímené 46
 podtržení 86
 podřezávání 7, 50
 pole 26, 28
 pomlčka 6, 49, 55
 popisek objektu 25, 82
 posouvání objektů vertikální 84
 poster 56
 postscript 43
 poznámka

 na okraj 62
 pod čarou 15, 16, 21, 62
 práce
 bakalářská 18
 diplomová 18, 56
 disertační 56
 ročníková 18
 semestrální 19
 seminární 19
 pravidlo typografické 1
 preambule 42, 50, 52, 57, 63, 65, 89
 prezentace 56
 procento 49
 projekt 66
 proložení znaků 5
 promile 50
 prostředí 44, 52, 53
 číslované 57
 plovoucí 82
 protitul 17
 předmluva 17, 61
 přesah 70
 příkaz 39
 příloha 18, 60
 půlčtverčík 44

R

rámeček 77, 83, 84
 čárkovaný 77
 registr délkový 52, 84
 rejstřík 18, 28, 61, 89
 rodina písma 45
 rovnice 50
 rozdělení slova 71
 rozložení dokumentu 3
 řádkování 10, 52
 přesně 10
 řez písma 5, 13, 45, 48, 57
 tapír 57

S

sazba
 vícesloupcová 66
 serif 5
 seznam 10
 autorů 61
 číslovaný 10, 24, 54, 55, 69

- dokumentový 26, 89
 literatury 18, 29, 61, 88
 obrázků 18, 27, 60, 89
 podle stylů 89
 podle titulků 27
 s popisky 10, 54, 55
 tabulek 18, 27, 60, 89
 víceúrovňový 10
 výčtový 10, 54
 skládání bloků 66
 skupina 44–47, 52
 hlavní 44
 konec 44, 46, 52, 53
 prázdná 49, 50
 slitek 7, 50
 sloupec 21
 soubor
 binární 34
 čistě textový 34
 spojovník 5, 49, 55, 71
 strana titulní 61, 62, 65
 struktura dokumentu 17, 65
 stránka
 lichá 17
 sudá 17
 stupeň 50
 písma 5, 13, 44, 45, 56
 styl 12, 13, 52, 56
 odstavcový 9, 52
 stránky 63
 vytvoření 57
 symbol 50
 krát 6, 50
 systém
 typografický 1
 šablona 12, 13, 16, 24, 35
 šířka
 bloku 67
 stránky 67
- T**
- tabulka 79, 82
 tabulátor 78
 tělo dokumentu 42
 T_EX 39, 40
 T_EXLive 41
 T_EXnicCenter 41, 42, 66
- tiráž 18
 tisk
 jednostranný 17
 oboustranný 17
 tiskárna virtuální 36
 titul 61
 hlavní 17
 titulek
 automatický 25
 objektu 25
 tmavost písma 45
 trojtečka 6, 49
 třída dokumentu 56, 59
 typ písma 4
 typografie 1
- U**
- úhel sklonu 75
 úvod 18
 uvozovky 6, 49
- V**
- varování 70
 verzálka 7
 vlna 51
 vlnovka 49, 86
 vložení
 nadpisu do obsahu 89
 textu do obsahu 90
 výpustka 6
 vysvětlivka 15, 21
- W**
- warning 70
 WYSIWYG 38, 43
- Z**
- záhlaví 21, 56, 63
 mrtvé 21
 živé 21
 záložka 23, 69
 zápatí 21, 56, 63
 zarovnání 52
 čísel 81
 objektů na stránce 84
 sloupců v tabulce 79
 závěr 18
 záznam

laboratorní	56
vydavatelský.....	17
zlom stránkový	10
znaménko	
diakritické	51
interpunkční	11
životopis	56

REJSTŘÍK PŘÍKAZŮ A PROSTŘEDÍ

Symbols

\'	78
\,	49
\-	71
\=	78
\>	78
\#	50
\\$	50
\%	49, 50
\&	50, 80
\\	52, 78, 80
\^	51
\'	78
11pt	65

A

a4paper	65
abstract	40, 61
\addcontentsline	60, 89, 90
\addtocontents	90
\ae	51
\alpha	50
\and	61
angle=	74
\appendix	60, 65
arabic	63, 64
\ast	50
\author	40, 61, 65

B

\backmatter	61, 62
\baselinestretch	52
bb=	74

\begin	42, 52
\bfseries	46, 58, 62, 90
\bibitem	88
\bigskip	49, 90

C

\c	51
\caption	82
\celsius	50
center	52, 53
\centering	53, 82
\cfoot	63, 64
\chapter	60, 65
\chapter*	60
\thead	63
\CheckedBox	54
\circle	75, 76
\circle*	75, 76
\cite	88, 89
\cline	80
\color	85, 86
\colorbox	85, 86
\copyright	50

D

\dashbox	75, 77
\date	40, 61, 65
\def	52, 57, 58
\definecolor	85–87
\degree	50
\depth	84
description	55
document	40

\documentclass...40, 44, 56, 57, 65
 \Dontwash.....50
 \dotfill.....49, 79, 80
 \dots.....49
 dvips.....87

E

em.....44, 52
 \emph.....40, 46, 88
 empty.....63, 64
 \end.....42, 52
 enumerate.....55
 ex.....44, 52

F

fancy.....63, 64
 \fancyfoot.....63, 64
 \fancyhead.....63, 64
 \fbox.....62, 67, 84, 85
 \fcolorbox.....85, 86
 figure.....82
 floatingfigure.....83
 floatingtable.....83
 flushleft.....53
 flushright.....53
 \font.....57
 \footnote.....62
 \footnotemark.....62
 \footnotesize.....45
 \footnotetext.....62
 \footrulewidth.....64
 \framebox.....75, 77, 84, 85
 \frontmatter.....61, 62

G

grey.....85, 86

H

headings.....63
 \headrulewidth.....64
 \height.....84
 height=.....74
 \hfill.....49, 67, 72
 \hline.....80
 \href.....87
 \hrulefill.....49, 58
 \Huge.....45
 \huge.....45, 62

\hypersetup.....87
 \hyphenation.....71

I

\include.....65
 \includegraphics.....72, 73, 83
 \includegraphics*.....72, 73
 \index.....89
 \input.....65
 \item.....54, 55
 \itemindent.....55
 itemize.....52, 54, 55, 58
 \itemsep.....55, 58
 \itshape.....46, 55, 72

K

\kill.....78

L

\label.....55, 69, 70, 82
 \labelsep.....55
 \LARGE.....45
 \Large.....45
 \large.....45, 72
 \LaTeX.....50
 \LaTeXe.....50
 \leftmargini.....55
 \leftmarginii.....55
 \leftmarginiii.....55
 \leftmarginiv.....55
 \leftmark.....64
 \lfoot.....63
 \lhead.....63, 64
 \line.....75, 76
 \listoffigures.....89
 \listoftables.....89
 lof.....60
 \lomitko.....52
 lot.....60

M

\mainmatter.....61, 62
 \makebox.....75, 77, 84, 85
 \makeindex.....89
 \maketitle.....40, 61, 65
 \marginpar.....62
 \markboth.....63
 \markright.....63

`\MartinVogel`.....50
`\mbox`.....84
`\mdseries`.....46
`\medskip`.....49
`minipage`.....62, 67, 79, 83–85
`multicols`.....67
`\multicolumn`.....80
`\multiput`.....75, 76
`myheadings`.....63

N

`\newcommand`.....57, 58
`\newenvironment`.....57, 58, 70
`\newline`.....58
`\newpage`.....52, 65, 89, 90
`\newtheorem`.....57, 58, 69, 70
`\noindent`.....68, 90
`\normalfont`.....46, 54, 55, 58, 70
`\normalsize`.....45, 52

O

`\oval`.....75, 76

P

`\pagecolor`.....85
`\pagenumbering`.....63, 64
`\pageref`.....69, 70, 82
`\pagestyle`.....63, 64
`\par`.....52, 57
`\paragraph`.....54
`\parbox`....67, 72, 75, 77, 79, 83, 84
`\parindent`.....52
`\parpic`.....83
`\parskip`.....52
`\part`.....60
`\partopsep`.....55
`picture`.....74, 75, 83, 84
`plain`.....63
`\poptabs`.....78
`\printindex`.....89, 90
`\pushtabs`.....78
`\put`.....74–76

Q

`\qbezier`.....76, 86
`\qqquad`.....49
`\quad`.....49, 58
`quotation`.....52

R

`\raggedleft`.....53, 62
`\raggedright`.....53
`\raisebox`.....62, 84, 85
`\ref`.....55, 69, 70, 82
`\renewcommand`.....57, 64
`\renewenvironment`.....57
`\resizebox`.....72
`\rfoot`.....63
`rgb`.....85, 86
`\rhead`.....63, 64
`\rightmark`.....64
`\Rightscissors`.....50
`\rmfamily`.....46
`Roman`.....63, 64
`roman`.....63
`\rotatebox`.....72, 74
`\rule`.....49, 50, 58, 67

S

`scale=`.....74
`\scalebox`.....72, 74, 83
`\scriptsize`.....45
`\scshape`.....46
`\section`.....40, 60
`\section*`.....60
`\setlength`.....52, 55, 57, 58
`\sffamily`.....46
`\shortstack`.....76, 77
`\slshape`.....46
`\small`.....45, 75
`\smallskip`.....49
`\subsection`.....60
`\subsubsection`.....60

T

`tabbing`.....78, 79
`table`.....82
`\tableofcontents`.....65, 89
`tabular`.....79
`\tapir`.....57
`\Telefon`.....50
`\TeX`.....50
`\textbf`.....46, 58
`\textcircled`.....58
`\textcolor`.....85, 86
`\textit`.....46

<code>\textmd</code>	46
<code>\textmusicalnote</code>	50
<code>\textonehalf</code>	50
<code>\textperthousand</code>	50
<code>\textsc</code>	46, 88
<code>\textsl</code>	46
<code>\texttildelow</code>	50
<code>\texttrademark</code>	50
<code>\texttt</code>	46
<code>\textup</code>	46
<code>\textwidth</code>	67, 68
<code>thebibliography</code>	88
<code>\thepage</code>	64
<code>\thispagestyle</code>	63, 90
<code>\times</code>	50
<code>\tiny</code>	45, 58
<code>\title</code>	40, 61, 65
<code>toc</code>	60
<code>\today</code>	40, 61, 64, 65
<code>\totalheight</code>	84, 85
<code>\ttfamily</code>	46

U

<code>\underline</code>	86
<code>\upshape</code>	46
<code>\url</code>	86–88
<code>\usepackage</code> ..	40, 48, 57, 58, 63, 65, 72, 86, 87
<code>\uv</code>	49

V

<code>\vector</code>	75, 76
<code>\vfill</code>	49
<code>\vspace</code>	49

W

<code>\width</code>	84, 85
<code>width=</code>	74
<code>wrapfigure</code>	83
<code>wratable</code>	83

REJSTŘÍK BALÍČKŮ, TŘÍD A VOLEB

Symbols

l1pt.....57

A

a0poster56

a4paper.....56, 57

a4wide.....58, 65

acmtrans2e.....56

adfathesis56, 59

amsart56

amsbook.....56

article40, 56, 57, 60, 61, 63, 89

B

beamer56, 59

book56, 60–63

C

captcont83

ccaption.....83

citecolor.....87

classicthesis56

clock59

color57, 85, 87

colorlinks87

comment.....58

csindex.....89

cv.....56

czech40, 58, 65

D

dvipdfm (program).....86

dvips.....72, 73, 85

E

eeepic77

elpres56

elsevier.....56

epic.....77

F

fancyhdr63

floatflt.....83

fncychap.....59

G

gensymb.....50, 58

graphics.....71–73

graphicx.....71, 73

H

hitec.....56

hyperref57, 86, 87

I

ifmslide56

index.....89

inputenc87

J

jastthesis59

L

labbook56

latin2.....87
 leaflet.....56
 letter.....56
 linkcolor.....87
 lncs.....56

M

makeidx.....89
 marvosym.....50, 58
 memoir.....56
 moderncv.....56
 multicol.....67

N

nature.....56

O

octavo.....56, 59

P

paper.....56
 parallel.....59
 pdftex.....72, 73, 85
 picins.....83
 pictex.....77
 picture.....85, 86
 pmgraph.....77
 prosper.....56, 59
 pstricks.....77

Q

qsm.....59
 quotchap.....59

R

refart.....56
 refrep.....56
 report.....56, 57, 60, 61, 63

S

scrartcl.....56
 scrbook.....56
 scrreprt.....56, 59
 sectsty.....59
 shapepar.....59
 slashbox.....81
 subfig.....83
 subfigure.....83
 subfloat.....83

supertabular.....81

T

tabularx.....81
 texpower.....56
 textcomp.....50, 58
 twocolumn.....66
 twoside.....56, 57

U

ua-thesis.....56
 unicode.....87
 url.....86
 urlcolor.....87

W

wasysym.....58
 wrapfig.....83

X

xypic.....77