

Dokumentace



České
vysoké
učení technické
v Praze

F3

Fakulta elektrotechnická
Katedra matematiky

CTUstyle – návod k použití

Šablona v plain \TeX u

pro sazbu studentských závěrečných prací na ČVUT

Petr Olšák

Leden 2013

<http://petr.olsak.net/ctustyle.html>

Poděkování / Prohlášení

Chtěl bych poděkovat své manželce Ludmile za podporu nejen finanční. Díky tomu mohu na svém pracovišti dělat, co mě baví, a nejsem stresován výplatní páskou.

Prohlašuji, že jsem předloženou práci vypracoval samostatně a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

V Praze dne 13. 13. 2013

.....

Abstrakt / Abstract

Tento dokument ukazuje a testuje použití (možná) oficiálně doporučené plainTeXové šablony CTUstyle pro sazbu bakalářských, diplomových a disertačních prací na Českém vysokém učení technickém v Praze. Šablona definuje všechny povinné strukturní elementy zmíněných závěrečných prací a formátuje jejich obsah tak, aby splňovala na škole daná formální pravidla.

Klíčová slova: styl dokumentu; šablona; bakalářská, diplomová, disertační závěrečná práce; TeX.

This document shows and tests an usage of the plainTeX officially (may be) recommended design style CTUstyle for bachelor (Bsc.), master (Ing.), or doctoral (Ph.D.) theses at the Czech Technical University in Prague. The template defines all thesis mandatory structural elements and typesets their content to fulfil the university formal rules.

Keywords: document design template; bachelor, master, Ph.D. thesis; TeX.

Title translation: CTUstyle – the user manual (the plainTeX template for theses at CTU)

/ Obsah

1 Úvod	1
1.1 Minimální dokument	1
1.2 Deklarační příkazy vymezující údaje o dokumentu	2
1.3 Rozvržení zdrojového textu dokumentu	3
1.4 Pracovní prostředí	4
1.5 Řešení problémů	4
2 Části dokumentu	6
2.1 Kapitoly, sekce, podsekce, přílohy	6
2.2 Povinné části závěrečné práce	6
2.2.1 Rozvržení dokumentu	6
2.2.2 Literatura	6
2.2.3 Zadání práce	7
2.3 Obrázky, tabulky, listingy a další	7
2.3.1 Obrázky	7
2.3.2 Tabulky	8
2.3.3 Automaticky číslované objekty	8
2.3.4 Listingy, výpisy kódů	9
2.3.5 Poznámky pod čarou	10
2.3.6 Zvýrazňování textů	10
2.3.7 Uvozovky, pomlčky, nezlomitelné mezery	10
2.3.8 Odkazy do internetu	10
2.3.9 Seznamy	11
2.3.10 Slovníček zkratek	11
2.3.11 Rejstřík	11
2.4 Citace na literaturu, poznámky k bibTeXu	11
2.5 Citace na literaturu podle normy ČSN ISO 690	12
2.6 Jak vložit přílohu A	12
2.7 Pracovní verze dokumentu	13
2.8 Teze práce	14
3 Poznámky k typografii	16
Literatura	17
A Zadání práce	19
B Zkratky a symboly	20
B.1 Zkratky	20
B.2 Symboly	20

Tabulky / Obrázky

1.1. Seznam deklarčních příkazů	3
2.1. Počet absolventů FEL	8

2.1. Ukázka vložení obrázku na střed	7
---	---

Kapitola 1

Úvod

Tento dokument popisuje užití plain \TeX ového makra CTUstyle, které možná bude oficiálně doporučeno na tvorbu bakalářských, magisterských a doktorských studentských závěrečných prací na Českém vysokém učení technickém v Praze. Zdrojový text tohoto dokumentu může posloužit jako příklad, jak vytvořit vlastní dokument.

Abyste mohli použít tuto šablonu, potřebujete mít základní znalost o \TeX u [1–2] a fungující \mathcal{CS} plain¹⁾ ve verzi aspoň Nov. 2012 s makrem OPmac²⁾ ve verzi aspoň Jan. 2013. Pro pokročilejší \TeX isty jsou k dispozici knihy [3–5]. \mathcal{CS} plain je součástí balíčku \TeX live³⁾ na většině linuxových distribucí. \TeX live je volně dostupný software a je možné jej instalovat i na Macintosh nebo MS Windows.

1.1 Minimální dokument

Můžete si vytvořit soubor `minimal.tex` s tímto obsahem:

```
\input ctustyle
\worktype [B/CZ]
\faculty {F3}
\department {Katedra permoníků}
\title {Minimální dokument}
\author {Pepa z Depa}
\date {Leden 2013}
\abstractEN {This document is for testing purpose only.}
\abstractCZ {Tento dokument je pouze pro potřeby testování.}
\declaration {Prohlašuji, že jsem se neflákal.}
\makefront

\chap Úvod

Text úvodu.

\sec Myšlenka

Další text.
\bye
```

Založte si pracovní adresář a do něj dejte šablonu `ctustyle.tex` a logo `ctulogo.pdf`. Dále tam uložte výše zmíněný soubor `minimal.tex` (včetně prázdných řádků za slovy Úvod a Myšlenka) a zpracujte tento soubor pdf \mathcal{CS} plainem. To znamená, že na příkazový řádek napíšete:

```
pdfcsplain minimal
```

¹⁾ <http://petr.olsak.net/csplain.html>

²⁾ <http://petr.olsak.net/opmac.html>

³⁾ <http://www.tug.org/texlive/>

Objeví se varování o chybějících údajích pro obsah. Zpracujte tedy tento soubor pdfCŠplainem ještě jednou. Nyní by už měla všechna varování zmizet a vytvoří se soubor `minimal.pdf`.

Co obsahuje minimální dokument? Na prvním řádku je zavedeno makro `CTUstyle`. Druhý řádek určuje typ dokumentu. Jsou k dispozici následující možnosti (vyberte jen jednu z nich):

```
\worktype [B/CZ] % bakalářka v češtině
\worktype [B/SK] % bakalářka ve slovenštině
\worktype [B/EN] % bakalářka v angličtině
\worktype [M/CZ] % diplomka v češtině
\worktype [M/SK] \worktype [M/EN] % diplomka ve slovenštině, angličtině
\worktype [D/EN] % disertace v angličtině
\worktype [D/CZ] \worktype [D/SK] % to snad asi ne :-)
\worktype [O/CZ] \worktype [O/SK] \worktype [O/EN] % ostatní typ práce
```

Dále v příkladu minimálního dokumentu následují tzv. *deklarační příkazy*, které vymezují základní údaje o dokumentu. Žádný z nich nesmí chybět, jinak `TEX` ohlásí chybu.

Pak následuje příkaz `\makefront`, který vytvoří titulní stranu, vypíše text prohlášení a texty abstraktů a také vytvoří obsah dokumentu (včetně případného seznamu obrázků a tabulek).

Za příkazem `\makefront` pokračuje text dokumentu. Kapitola je zahájena příkazem `\chap` a sekce příkazem `\sec`. Za těmito příkazy následuje název kapitoly, resp. sekce, který musí být oddělen od dalšího textu prázdným řádkem.

1.2 Deklarační příkazy vymezující údaje o dokumentu

Kromě výše zmíněných povinných deklaračních příkazů je obvyklé použít i některé další. Všechny tyto příkazy jsou uvedeny v tabulce 1.1. Není-li některý z nepovinných deklaračních příkazů v dokumentu uveden, znamená to, že odpovídající údaj je prázdný. Na pořadí deklaračních příkazů v dokumentu nezáleží, ale všechny musejí být umístěny před příkazem `\makefront`.

Za každým deklaračním příkazem následuje údaj uzavřený do složených závorek `{...}`. Údaje píšete v hlavním jazyku dokumentu, tj. v jazyku, který jste vyplnili v parametru příkazu `\worktype`. Pouze údaje ve tvaru `\slovoEN` resp. `\slovoCZ` resp. `\slovoSK` vyplňujete v jazyce podle zkratky v názvu příkazu.

Mezi povinné údaje patří `\abstractEN` a dále jeden ze dvou `\abstractCZ` nebo `\abstractSK`. I když je hlavním jazykem dokumentu angličtina, vyžaduje se uvést ještě abstrakt v místním nebo mateřském jazyce.

Není-li hlavním jazykem dokumentu angličtina, doporučuje se uvést údaj `\titleEN`, resp. `\subtitleEN`, s překladem názvu, resp. podnázvu, do angličtiny. A obráceně: je-li hlavním jazykem angličtina, je samozřejmě údaj `\title` v tomto jazyce. Pak se ale doporučuje uvést ještě údaje `\titleCZ`, `\subtitleCZ` nebo `\titleSK`, `\subtitleSK`. Tyto údaje se objeví pod abstraktem v daném jazyce.

Je-li titul příliš dlouhý, pak se na titulní straně pravděpodobně rozlomí do více řádků. Někdy ale by si člověk přál jiné rozlomení, než vymyslel stroj, protože je vhodnější rozdělit název do řádků podle významových celků, kterému stroj nemůže rozumět. V takovém případě můžete do názvu do místa, kde si přejete provést zlom, přidat příkaz `\nl` (zkratka za new line).

Typ dokumentu (B, M, D) se v hlavním jazyce vytiskne na titulní straně jako „bakalářská práce“, „diplomová práce“ atd. Pokud tvoříte jiný typ dokumentu, píšete

Příkaz	Význam
<code>\faculty *</code>	Fakulta. Možnosti: F1, F2, F3, ..., F8, MUVS
<code>\department *</code>	Název katedry nebo ústavu.
<code>\title *</code>	Titul dokumentu.
<code>\subtitle</code>	Doplňkový titul dokumentu.
<code>\author *</code>	Jméno a příjmení autora.
<code>\supervisor</code>	Jméno a tituly vedoucího práce (školitele u Ph.D.).
<code>\date *</code>	Měsíc a rok vytvoření práce.
<code>\authorinfo</code>	email autora nebo něco podobného.
<code>\studyinfo</code>	Název studijního programu, studijního oboru.
<code>\workinfo</code>	URL práce, loga sponzorů nebo něco podobného.
<code>\workname</code>	Jméno typu práce, je-li typu „ostatní“.
<code>\titleEN</code>	Titul práce, není-li <code>\title</code> v angličtině.
<code>\titleCZ</code>	Titul práce, není-li <code>\title</code> v češtině.
<code>\titleSK</code>	Analogie jako <code>\titleCZ</code> pro slovenštinu.
<code>\subtitleEN</code>	Doplňkový titul, není-li <code>\subtitle</code> v angličtině.
<code>\subtitleCZ</code>	Doplňkový titul, není-li <code>\subtitle</code> v češtině.
<code>\subtitleSK</code>	Analogie jako <code>\subtitleCZ</code> pro slovenštinu.
<code>\pagetwo</code>	Text umístěný na straně dva za titulní stránkou dole.
<code>\abstractEN *</code>	Abstrakt v angličtině.
<code>\abstractCZ *</code>	Abstrakt v češtině.
<code>\abstractSK</code>	Abstrakt ve slovenštině (nahrazuje <code>\abstractCZ</code>).
<code>\keywordsEN</code>	Klíčová slova v angličtině.
<code>\keywordsCZ</code>	Klíčová slova v češtině.
<code>\keywordsSK</code>	Klíčová slova slovensky (nahrazuje <code>\keywordsCZ</code>).
<code>\thanks</code>	Text poděkování.
<code>\declaration *</code>	Text prohlášení.
<code>\specification</code>	Zadání práce, viz sekci 2.6.

Tabulka 1.1. Seznam deklaračních příkazů. Hvězdičkou jsou označeny povinné příkazy.

`\worktype[0/CZ]` (nebo jiný jazyk). V takovém případě jsou deklarační příkazy `\declaration`, `\abstractEN`, `\abstractCZ/SK` nepovinné a typ práce je naopak nutné vymezit deklaračním příkazem `\workname`. Například:

```
\workname {Domácí úkol do předmětu AB004C}
```

Doporučuji vyjít ze zdrojového souboru tohoto dokumentu `ctustyle-doc.tex` a zkopírovat si ho do výchozího souboru vaší práce, například `bakalarka.tex`. Pak v tomto souboru udělejte změny na místech, kde se vaše práce liší od tohoto dokumentu. Svou práci pak můžete T_EXovat příkazem:

```
pdfcsplain bakalarka
```

1.3 Rozvržení zdrojového textu dokumentu

Je na autorovi, zda se rozhodne za příkazem `\makefront` pokračovat ve psaní a vloží celý svůj dokument do jediného zdrojového souboru (podobně jako zde minimální dokument). V případě rozsáhlejší práce ovšem je někdy výhodné rozdělit dokument napří-

klad podle kapitol na jednotlivé soubory a každou kapitolu načítat jako soubor zvlášť příkazem `\input`. Tato koncepce je předvedena v souboru `ctustyle-doc.tex`.

Čtenářům doporučuji podívat se do zdrojových souborů tohoto dokumentu `uvod.tex`, `popis.tex` a `prilohy.tex`, aby se seznámili s tím, jak vypadá „kultura psaní zdrojových souborů“. Je třeba se vyhnout tomu, aby to bylo zmatené tak, že se v tom nikdo nevyzná, třebaže je \TeX dosti tolerantní. Podobnost s jinými programovacími jazyky není čistě náhodná.

Pokud jste si důkladně prohlédli soubor `ctustyle-doc.tex`, jistě jste si všimli, že je tam použit znak `%` jako komentářový znak, který zahajuje komentář (až do konce řádku). Komentáře jsou při zpracování dokumentu ignorovány. Můžete se zaměřit při zpracování dokumentu jen na jednu kapitolu, pokud ostatní příkazy `\input soubor` zakryjete procentem. To je obvyklá praxe, když se člověk soustředí na jeden menší problém a nechce se zdržovat zpracováním celého dokumentu.

1.4 Pracovní prostředí

Textový editor, ve kterém pořizujete zdrojový text dokumentu, je volbou každého autora. V této věci zde nenajdete žádné doporučení. Prohledejte internet, zeptejte se kolegů. Editor samozřejmě musí být „programátorský editor“, tj. nesmí si vymýšlet žádné skryté formátovací elementy. Soubor musí být uložen na disk jako prostý text v kódování češtiny nebo slovenštiny v UTF-8. Výhodné je, pokud editor disponuje možností použít klávesovou zkratku na spuštění externího příkazu, v tomto případě příkazu `pdfcsplain`. Ve vedlejším okně můžete mít trvale spuštěný prohlížeč PDF dokumentů konfigurovaný tak, aby dokázal poznat, že došlo ke změně PDF dokumentu a obnovil v takovém případě zobrazení dokumentu.

Dokument je rozumné zpracovávat včetně vkládaných obrázků (ve formátu JPG, PNG nebo PDF) v jednom vyhrazeném adresáři. \TeX musí být také schopen dohledat soubory se šablonou `ctustyle.tex` a `ctulogo.pdf`. To nejjednodušeji zajistíte tak, že tyto soubory budou přítomny také v pracovním adresáři dokumentu. Není to ale nutné. Můžete je nakopírovat do \TeX ové distribuce, tj. uložit je na vhodné místo (podle návodu k \TeX ové distribuci) a spustit obnovovací příkaz typu `texhash`. Pak tyto soubory \TeX dokáže dohledat a nemusíte je mít v aktuálním adresáři.

1.5 Řešení problémů

Je možné, že budete mít instalovanu starší distribuci \TeX live, která obsahuje verzi \LaTeX plainu starší než Nov. 2012. Požadovaná verze \LaTeX plainu je v distribuci \TeX live teprve od prosince 2012. U starší verze byste museli přegenerovat formát `pdf\text{\LaTeX}plain`, aby uměl číst UTF-8 kódování, a instalovat soubory `opmac.tex` a `ams-math.tex` manuálně ze stránky OPmac¹). Doporučuji použít verzi \LaTeX plainu Jan. 2013, která opravuje některé chyby zjištěné koncem roku. Soubory z aktuální verze získáte na [www stránce \text{\LaTeX}plainu²](http://www.strance\text{\LaTeX}plainu2)).

V lednu byly opraveny též některé chyby v makru OPmac. Je nutné instalovat makro OPmac ve verzi aspoň Jan. 2013.

Soubor maker CTUstyle je zatím v testovacím provozu. Není vyloučeno, že se objeví nějaké chyby nebo záhadné chování. Máte-li pocit, že se tak stalo, zašlete prosím na

¹) <http://petr.olsak.net/opmac.html>

²) <http://petr.olsak.net/csplain.html>

adresu `petr@olsak.net` popis chyby a připojte k němu `.log` soubor dokumentu a pokud možno minimální ukázkový dokument, na kterém je chyba patrná.

Autor CTUstyle je připraven prostřednictvím emailu poradit jakémukoli studentu ČVUT se záludnostmi při formátování závěrečné práce, pokud pro formátování použil `CSplain` (tedy nikoli `LATEX`).

Kapitola 2

Části dokumentu

Tento dokument nemůže sloužit jako návod k použití \TeX u a \LaTeX u. Domnívám se ale, že metodou analogie je schopen i \TeX ový nováček vytvořit obvyklý dokument. Doporučuji mu prostudovat stručný text [1] a uživatelskou dokumentaci k makru `OPmac` `opmac-u.pdf` [6]. Na jednotlivé odstavce této dokumentace k `OPmac` budu v této kapitole odkazovat. Nově je též k dispozici text o základech plain \TeX u [2].

2.1 Kapitoly, sekce, podsekce, přílohy

Dokument lze členit na kapitoly, sekce, podsekce a přílohy. Používají se k tomu příkazy vysvětlené v dokumentaci k `OPmac` v sekci třetí, Kromě příkazů z `OPmac` makro `CTUstyle` přidává příkaz `\app`. Takže seznam příkazů pro vymezení základní struktury dokumentu vypadá takto:

```
\chap Nadpis kapitoly <ukončený prázdným řádkem>
\sec Nadpis sekce <ukončený prázdným řádkem>
\secc Nadpis podsekce <ukončený prázdným řádkem>
\app Nadpis přílohy <ukončený prázdným řádkem>
```

Přílohy uvozené příkazem `\app` se chovají stejně jako kapitoly, jen nejsou číslovány, ale jsou k nim automaticky vzestupně přiřazena písmena A, B, C... Také přílohy mohou být členěny na sekce a podsekce.

2.2 Povinné části závěrečné práce

2.2.1 Rozvržení dokumentu

Je doporučeno číslované kapitoly zahájit úvodem, pak další kapitoly podle potřeby práce a poslední kapitola by měla být označena jako závěr. Následně musí být zařazen seznam literatury (nečíslovaná kapitola) a pak povinná příloha A se zadáním práce (u bakalářské a diplomové práce) a následně případné další přílohy. Mezi těmito dalšími přílohami se velmi doporučuje zařadit seznam zkratk a symbolů, jako zde v příloze B.

2.2.2 Literatura

Každá studentská závěrečná práce musí obsahovat seznam použitých zdrojů. Toto je jediná nečíslovaná kapitola umístěná na konci textu práce ale před přílohami. `CTUstyle` nabízí příkaz `\bibchap`, který je potřeba použít místo příkazu `\chap`. Příkaz vytvoří záznam pro obsah, vytiskne slovo Literatura (nebo References v anglicky psané práci). Toto slovo se generuje automaticky, příkaz zapisujte bez parametru.

Způsob tvorby seznamu literatury a způsob citace na literaturu v textu je podrobněji vyložen v sekci 2.4 nebo 2.5.

2.2.3 Zadání práce

Každá bakalářská nebo diplomová závěrečná práce musí obsahovat úplný text zadání práce. Doporučuji jej zařadit do přílohy A (jako v tomto dokumentu) a označit ji názvem „Zadání práce“ (anglicky Specification). Někdy je v požadavcích uvedeno vložit zadání práce hned za titulní stranu. Ačkoli to z typografického pohledu nedoporučuji, CTUstyle to umožňuje. Podrobnější komentář k této problematice je v sekci 2.6

2.3 Obrázky, tabulky, listingy a další

2.3.1 Obrázky

Obrázky ve formátu JPG, PNG (bitmapové) nebo PDF (vektorové i bitmapové) je možné vložit příkazem `\inspic`. Viz sekci 12 v dokumentaci k OPmac. Pod obrázek je nutné vložit popisek příkazem `\caption/f`, viz sekci 4 v dokumentaci k OPmac. CTUstyle navíc definuje příkaz `\cinspic`, který umístí obrázek doprostřed. Například:

```
\medskip
\picw=5cm \cinspic ctulogo.pdf
\caption/f Ukázka vložení obrázku na střed, což je asi nejobvyklejší.
\medskip
```

vytvoří:



Obrázek 2.1. Ukázka vložení obrázku na střed, což je asi nejobvyklejší.

Makro `\cinspic` vyžaduje jméno souboru s příponou ukončené mezerou podobně jako příkaz `\input`.

Pokud se obrázek vyskytuje dole na stránce tak, že stranu přeplní, nastávají problémy se stránkovým zlomem. Proto je vhodné obrázky vložit i s popisem do dvojice příkazů `\midinsert` a `\endinsert`. V takovém případě obrázek implicitně zůstane, kde je, ale při potížích odpluje na začátek následující stránky:

```
\midinsert
\picw=5cm \cinspic ctulogo.pdf
\caption/f Ukázka vložení obrázku na střed, což je asi nejobvyklejší.
\endinsert
```

Místo příkazu `\midinsert` můžete použít `\topinsert`. V takovém případě obrázek odpluje na horní část stránky vždy. Raději má vršek aktuální stránky, ale když to není možné, odpluje na stránku následující.

2.3.2 Tabulky

Pro tabulky použijte příkaz `\ctable{<deklarace>}{<data>}`, který je odvozen z příkazu `\table` dokumentovaného v OPmac v sekci 11. CTUstyle definuje `\ctable` tak, že tabulku navíc podkládá modrým pozadím (což je součástí typografického návrhu šablony) a centruje ji. Například:

```
...vytvoří tabulku~\ref[absolventiFEL].

\midinsert \clabel[absolventiFEL]{Počet absolventů FEL ČVUT}
\ctable{!rrrrr}{
\hfil number of      & 2007 & 2008 & 2009 & 2010 & 2011 \crl \tskip4pt
students Bc. and Mgr. & 6313 & 5913 & 5951 & 5188 & 4737 \cr
graduate Bc. and Mgr. & 1195 & 1489 & 1379 & 1160 & 1260 \cr
students Ph.D.        & 457  & 468  & 366  & 395  & 434  \cr
graduate Ph.D.        & 65   & 60   & 55   & 54   & 51   \cr
}
\caption/t Počet absolventů FEL ČVUT. Tabulka je převzata z~\cite[zyka].
\endinsert
```

vytvoří tabulku 2.1.

number of	2007	2008	2009	2010	2011
students Bc. and Mgr.	6313	5913	5951	5188	4737
graduate Bc. and Mgr.	1195	1489	1379	1160	1260
students Ph.D.	457	468	366	395	434
graduate Ph.D.	65	60	55	54	51

Tabulka 2.1. Počet absolventů FEL ČVUT. Tabulka je převzata z [7].

Doporučuji nerámovat tabulky do dalších rámečků, ale využít toho, že tabulka je automaticky ohraničena modrým podkladem. Je vhodné pouze vložit linku mezi záhlaví a údaje v tabulce (viz příkaz `\crl` v ukázce).

Tabulky (podobně jako obrázky) je vhodné zapouzdřit do dvojice příkazů `\midinset` a `\endinsert` nebo `\topinsert` a `\endinsert`.

2.3.3 Automaticky číslované objekty

Jak je možné si všimnout, CTUstyle automaticky čísluje kapitoly, sekce, podsekce, dodatky, tabulky, obrázky a pokud uživatel použije `\eqmark`, očíslovuje i rovnice. O tomto číslování a o odkazech na tato čísla v textu pojednává odstavec 4 v dokumentaci k OPmac. Zde jen stručně uvádím, že číslované objekty je potřeba označit interním ležbílíkem příkazem `\label` a pak je možné na ně odkazovat příkazem `\ref`. Existuje ještě možnost odkazovat na stránku příkazem `\pgref` a na literaturu příkazem `\cite`.

Kapitoly se číslují od jedné v celém dokumentu, sekce se číslují druhým číslem v pořadí od jedné v každé kapitole a podsekce se číslují třetím číslem od jedné v každé sekci. Hlubší zanoření (podpodsekce) není podporováno a není pro studentské práce doporučeno.

Tabulky se číslují od jedné v každé kapitole a obrázky (nezávisle na tabulkách) taky. Rovněž rovnice se číslují od jedné v každé kapitole. CTUstyle volí kompromis mezi krátkým číslováním (Tabulka 27) a dlouhým číslováním (Tabulka 2.4.6). První extrém nedává představu o kapitole, ve které je tabulka umístěna, a druhý extrém se čtenáři obtížně pamatuje.

CTUstyle definuje kromě příkazu `\label` ještě příkaz `\clabel[⟨lejblík⟩]{⟨text⟩}`, který funguje jako `\label[⟨lejblík⟩]`, ale navíc vloží takto označenou tabulku nebo obrázek do seznamu tabulek nebo obrázků. Tyto seznamy se vygenerují hned za obsahem dokumentu. Pozor: není-li tabulka nebo obrázek označen pomocí `\clabel`, v příslušném seznamu se neobjeví. Někoho může napadnout otázka, proč má psát `⟨text⟩` dvakrát: jednou pro seznam obrázků či tabulek v příkazu `\clabel` a jednou pod obrázek v příkazu `\caption`. Je to proto, že ty texty se mohou lišit. Typicky v obsahu budou stručnější. Ukázka použití `\clabel` je u výpisu kódu k tabulce 2.1.

CTUstyle umožňuje použít automaticky číslované odstavce. Je připraveno pět nezávislých čítačů označených A, B, C, D a E, každý z nich začíná v každé kapitole číslovat od jedné. Makro `\numberedpar{čítač}{slovo}` zahájí číslovaný odstavec ve tvaru `⟨slovo⟩ ⟨číslo kapitoly⟩.⟨hodnota čítače⟩`. Následující příklad deklaruje věty a důsledky číslované společnou řadou čísel a dále nezávisle číslované definice a příklady.

```
\def\veta      {\numberedpar A{Věta}}
\def\dusledek  {\numberedpar A{Důsledek}}
\def\definice  {\numberedpar B{Definice}}
\def\priklad   {\numberedpar C{Příklad}}
```

Po této deklaraci můžete psát `\definice` Necht $\$M\$$ je naprázdňá ... a objeví se odstavec zahájený takto:

Definice 2.1. Necht M je neprázdná ...

Další definice v této kapitole bude mít číslo 2.2, další 2.3 atd. K tomu mohou být přidány věty a důsledky číslované 2.1, 2.2, atd. Konečně i příklady v této kapitole budou číslovány 2.1, 2.2, atd. Před takto označené odstavce lze psát `\label[⟨lejblík⟩]` a dá se pak na ně odkazovat pomocí `\ref[⟨lejblík⟩]` a `\pgref[⟨lejblík⟩]`, tedy odkazování je stejné jako u všech ostatních automaticky číslovaných objektů.

■ 2.3.4 Listingy, výpisy kódů

Pro listingy, tj. výpisy kódu, použijte dvojici příkazů `\begtt` a `\endtt`, jak o tom píše dokumentace k OPmac v sekci 10. CTUstyle definuje `\tthook` tak, aby byly listingy podbarveny světle modrou barvou, což je součást grafického stylu.

Listingy se lámou do více stránek a jsou tištěny strojopisem, aby to navodilo atmosféru pohledu do textového programátorského editoru, který rovněž používá písmo s pevnou šířkou všech znaků. Pravda, atmosféru to nevytvoří dokonalou, protože textové editory dnes navíc používají prostředky pro zvýraznění některých slov (klíčových slov programovacího jazyka atd.). Toto CTUstyle nepodporuje. Chcete-li tedy navodit dokonalou atmosféru, uložte si zobrazení svého textového editoru jako obrázek a do dokumentu vložte obrázek. Ovšem listingy pomocí `\begtt` a `\endtt` jsou velmi doporučené, protože modrý podklad graficky ladí s celkovým návrhem CTUstyle.

Pokud chcete přímo v odstavci uvádět kusy kódů, obalte je do dvojice znaků "...". Je možné tedy psát třeba toto:

```
Chcete-li zdůraznit slovo, použijte {\em kurzívu}, do které
přepnete příkazem "\em", tedy "{\em zvýrazněné slovo}".
```

Tyto kusy kódu budou uvnitř odstavce tištěny strojopisem a nebudou podléhat řádkovému zlomu. Bohužel dvojice znaků "... " je možné použít jen uvnitř „obyčejného“ odstavce, nikdy nefungují uvnitř parametrů jiných příkazů (nadpisy kapitol, obsahy tabulek, atd.). V takových místech musíte do strojopisu přepnout explicitně pomocí `{\tt text}` a pohlídat si sazbu T_EXovsky citlivých znaků. Místo backslashe je možné psát `\slash` a místo procenta `\percent`.

2.3.5 Poznámky pod čarou

Pro poznámky pod čarou používejte `\fnote{<text>}` jak je popsáno v sekci 14 dokumentace k OPmac. Vytvoří to poznámku¹⁾. Poznámky pod čarou jsou číslovány na každé stránce od jedné. Doporučuji s takovými poznámkami šetřit.

Poznámky na okraji `\mnote`, o kterých také hovoří dokumentace k OPmac, nejsou při použití CTUstyle doporučeny.

2.3.6 Zvýrazňování textů

Základní text je psán antikvou (písmem Latin Modern odvozeným z Computer Modern). Chcete-li zdůraznit slovo, použijte *kurzívu*, do které přepnete příkazem `\em`, tedy `{\em zvýrazněné slovo}`. Je to obvyklý způsob zdůrazňování, který je typograficky vhodný, protože netrčí z textu, ale je viditelný při čtení.

Pokud chcete zdůraznit něco, aby to bylo **vidět z dálky**, použijte přepínač `\bf`, který při použití CTUstyle přepíná do tučného fontu bez serifů (tj. bez patek). Tedy `{\bf takto}`. V tomto fontu jsou řešeny i nadpisy.

Vyznačování podtrháním textu nebo prostrkáním nedoporučuji.

2.3.7 Uvozovky, pomlčky, nezlomitelné mezery

České uvozovky vypadají „takto“, **anglické** “takto”. V závislosti na jazyce použijte správné uvozovky. Můžete je napsat přímo v textovém editoru (v UTF-8 kódování), nebo T_EXovsky to uděláte `\uv{takto}` pro češtinu a ‘‘takto’’ pro angličtinu.

Pomlčky v typografii jsou dvě.

- Střední pomlčka: – (používá se bez mezer kolem ve významu „až“ nebo s mezerami jako pomlka ve větě).
- Dlouhá pomlčka: — (používá se v anglickém textu).

Můžete tyto znaky napsat přímo v editoru v UTF-8 kódování nebo T_EXovsky: -- (střední pomlčka), --- (dlouhá pomlčka). Čtenář vašeho textu vám strhne nemilosrdně body, pokud ve významu pomlčky použijete spojovník. Vypadá takto: „–“ a promění se na něj singl znak „-“ ve zdrojovém textu.

Nezlomitelná mezera je mezislovní mezera, ve které nedojde k zalomení do řádků. V T_EXových zdrojových textech se typicky tato mezera značí vlnkou „~“. Existuje program *vlna*²⁾, který dokáže zaměnit normální mezery za tyto vlnky ve zdrojovém textu za všemi výskyty neslabičných předložek, kam skutečně patří v češtině i slovenštině nezlomitelná mezera. O to se tedy uživatel při psaní textu nemusí starat, jen při závěrečných korekturách použije program *vlna* na všechny vstupní soubory se zdrojový textem a spustí T_EX znovu. Program *vlna* ovšem nedává vlnky před čísla citací a referencí a na mnoho míst, kam podle zvyklostí v sazbě taky patří. To si musí uživatel pohlídat sám.

2.3.8 Odkazy do internetu.

Do internetu by se nemělo odkazovat přímo v textu, ale pomocí poznámky pod čarou `\fnote`. Aby se stalo URL klikatelné a bylo vytištěno správně strojopisem, je nutno je vložit do parametru příkazu `\url`, tedy třeba `\url{http://petr.olsak.net}` vytvoří `http://petr.olsak.net`. Ovšem navíc je potřeba tento text poslat do poznámky pod čarou. CTUstyle definuje zkratku `\urlnote{<URL text>}`, která je totožná s `\fnote{\url{<URL text>}}`. Takže text z předchozího odstavce byl napsán takto:

¹⁾ Jako je tato.

²⁾ `ftp://math.feld.cvut.cz/olsak/vlna/`

Existuje program "vlna" `\urlnote{ftp://math.feld.cvut.cz/olsak/vlna/}`, který dokáže...

2.3.9 Seznamy

Tvorba seznamů s odrážkami je popsána v sekci 5 v dokumentaci k OPmac (příkazy `\begitems` a `\enditems`). Implicitní odrážku v seznamu definuje CTUstyle jako modrý poněkud zaoblený čtvereček. Podívejte se, jak to vypadá, do sekce 2.3.7 do místa, kde se mluví o pomlčkách. Pokud chcete použít seznam v seznamu, pro vnitřní seznam použijte `\style x`, což vytvoří poněkud menší modré tečky. Pro číslované seznamy použijte `\style n`.

2.3.10 Slovníček zkratk

Tato možnost je zařazena do CTUstyle od verze May 2014. Je možné si například do souboru `glosdata.tex` připravit následující obsah

```
\glos {ČVUT} {České vysoké učení technické v Praze}
\glos {FEL} {Fakulta elektrotechnická ČVUT}
\glos {UK} {Univerzita Karlova}
\glos {MFF} {Matematicko-fyzikální fakulta UK}
```

a zařadit jej do dokumentu na jeho začátek (nejlépe před `\makefront`) pomocí `\input glosdata`. Tím se ještě nic nestane. Nyní ale můžete někam do dokumentu napsat třeba

```
\app Slovníček\par \makeglos
```

a v uvedeném místě se objeví slovníček sestavený z `glosdata.tex` a uspořádaný podle abecedy, třebaže `glosdata` uspořádána dle abecedy nejsou. Chcete-li vypnout abecední řazení, pište na začátek dokumentu `\let\dosorting=\relax..`

Objeví-li se zkratka ze slovníčku někde v dokumentu, můžete ji označit pomocí příkazu `\glref`, například `\glref{ČVUT}`, a v tomto místě se vytvoří hypertextový odkaz do slovníčku.

Je též možné místo `\glref` použít jiné makro `\glosref{<zkratka>}{<význam>}`. To využijete tehdy, pokud *nechcete* mít souhrnný soubor `glosdata.tex`, ale chcete významy jednotlivých zkratk zapsat až v místě jejich výskytu. To má ale jistá omezení. Zatímco zkratka označená `\glref` se v dokumentu může vyskytovat vícekrát, její význam musí být deklarován právě jednou a deklarace významu všech zkratk musejí předcházet místu, kde je slovníček vytištěn pomocí `\makeglos`.

Chcete-li slovníček sestávající z více podsekcí, je nutné jej sestavit manuálně, jako například v příloze B tohoto dokumentu.

2.3.11 Rejstřík

Rejstřík se u studentských závěrečných prací nevyžaduje. Nic ale nebrání jej vytvořit a postupovat přitom podle odstavce 7 v dokumentaci k OPmac.

2.4 Citace na literaturu, poznámky k bibT_EXu

Odkazy v textu vytváříme příkazem `\cite[<lejblik>]`. Lejbliků může být v hranaté závorce více a jsou odděleny čárkou. Podrobněji je tato problematika popsána v manuálu k OPmac [6] v sekci 15.

Seznam použité literatury má být řazen podle pořadí odkazů na citace v textu. Toto je součástí zadání, viz přílohu A. Osobně se mi to jeví jako nerozumné rozhodnutí, ale zadání je třeba ctít. Použijete-li BibTeX, je doporučeno použít BibTeXový styl `plain`, postupovat dle dokumentace k OPmac a vytvořit přílohu se seznamem literatury takto:

```
\bibchap
\usebb1/c mybase
```

Asi nejpohodlnější možností k vytvoření seznamu literatury je přímé čtení databázového souboru `.bib` makry T_EXu bez použití bibT_EXu. Toto je nová možnost (od dubna 2014). Stačí si připravit `.bib` soubor s odpovídajícími údaji (například `mybase.bib`), přidat do záhlaví dokumentu příkaz `\input opmac-bib` a do místa, kam má být vložen seznam literatury, napsat

```
\bibchap
\usebib/c (simple) mybase
```

Můžete vyjít z existujícího souboru `mybase.bib` a přidat si tam další položky. Je tedy možné postupovat metodou analogie. Místo stylu `simple` je možné použít též styl `iso690`.

Upozornění: přímé čtení `.bib` souboru vyžaduje OPmac ve verzi aspoň Apr. 2014 a k němu soubory `opmac-bib.tex` a `opmac-bib-simple.tex`, které jsou dostupné na¹⁾. Při přechodu na tuto novější verzi OPmac si laskavě smažte pomocné `.ref` soubory, protože jejich přítomnost v aktuálním adresáři může způsobit chybu v překladu.

2.5 Citace na literaturu podle normy ČSN ISO 690

Jak si pozorný čtenář mohl všimnout, předchozí sekce popisuje tvorbu seznamu literatury, který není v souladu s normou [8]. Nevím o BibTeXovém stylu, který by respektoval tuto normu z roku 2007. Navíc tato norma doporučuje vkládat do záznamů česká slova, (například „dostupné z“, „vydání“, spojku „a“ mezi autory), což tedy není použitelné v případě, že je práce napsaná v angličtině.

Pokud přesto vedoucí práce trvá na dodržení normy, doporučuji následující postup. Seznam literatury si vygenerujte na²⁾. Od dubna 2014 nabízejí na těchto stránkách na můj podnět export do T_EXu. Uložte tento export do `bb1` souboru a případně pozměňte české fráze, píšete-li práci v angličtině³⁾. Takto připravený seznam literatury můžete načíst do dokumentu příkazem `\usebb1/c <jméno souboru>`. Do dokumentu budou zařazeny jen citované publikace a v pořadí podle citování.

Další možností je přímé čtení `.bib` souboru příkazem `\usebib` se stylem `iso690`.

2.6 Jak vložit přílohu A

Předpisy požadují vložit přílohu „Zadání práce“ (v případě bakalářky a diplomky) do jednoho svázaného výtisku jako originál (s podpisy a razítkem) a do dalších výtisků resp. do PDF vložit kopii tohoto formuláře. Doporučuji tedy tento formulář oskenovat a oříznout mu bílé okraje co nejvíce to jde. To lze udělat například v programu Gimp. Tím vznikne obrázek například `zadani.jpg`. Přílohy pak zahajete takovým kódem:

¹⁾ <http://petr.olsak.net/opmac.html>

²⁾ <http://www.citace.com>

³⁾ Frázi „dostupné z:“ → „available from“, spojku „a“ (bez čárky před) → „and“ (s čárkou před) atd.

```

\app Zadání práce

\picw=\hsize % obrázek na šířku sazby
\cinspic zadani.jpg
\nextoddpage

\app Další příloha

```

Makro CTUstyle zahajuje vždy první přílohu (tj. přílohu A) na liché (pravé) straně. Za konec poslední kapitoly může tedy kvůli tomu vložit prázdnou stranu (neboli vakát). Na první straně přílohy se tedy objeví nadpis: „Příloha A / Zadání práce“ graficky upravený podle CTUstyle. Pod ním bude oskenovaný formulář se zadáním práce. Pak příkazem `\nextoddpage` dáváte najevo, že další příloha bude začínat znovu na pravé stránce, takže vlevo se vloží vakát. Tím pádem bude příloha A zaujímat při duplexním tisku kompletní jeden list papíru v závěrečné práci.

Nyní stačí tento list papíru v jednom výtisku zaměnit za originální formulář a nechat svázat. Originál tedy nebude mít nadpis přílohy A v jednotném grafickém stylu, ale nemůžeme chtít všechno. Kopie už budou mít jednotný grafický styl.

Pokud předpisy bezpodmínečně požadují vložit zadání práce jako druhý list hned za titulní stranu, je možné využít deklarační příkaz `\specification`, který zařadíte mezi ostatní deklarační příkazy a napíšete tam:

```

\specification {\picw=\hsize \cinspic zadani.jpg }

```

Tato deklarace způsobí, že sken zadání (bez dalších textů) se umístí na stranu třetí (strana druhá je typicky vakát za titulem, pokud není deklarován `\pagetwo`). Strana čtvrtá se stane dalším vakátem a od strany páté se zobrazí poděkování/prohlášení atd. Navíc na stranu třetí bude odkazovat strukturovaný obsah PDF prohlížeče názvem Zadání, resp. Specification.

Máte-li dvě varianty zadání (např. ve dvou jazycích), je možné psát:

```

\specification {\picw=\hsize \cinspic specifi-en.jpg
               \vfil\break \cinspic specifi-cz.jpg }

```

V tomto příkladě se `specifi-en.jpg` zobrazí na straně třetí a `specifi-cz.jpg` na straně čtvrté. A stejně jako prve od strany páté pokračuje poděkování/prohlášení atd.

2.7 Pracovní verze dokumentu

Příkazem `\draft` vloženým před příkaz `\makefront` vznikne verze dokumentu označená datem vzniku a slovem Draft na každé stránce. Je to tedy pracovní (nefinální) verze.

V pracovní verzi jsou dále červeně vypsány lejbíky, které jste do dokumentu vložili pomocí `\label` nebo `\clabel`. Jsou umístěny v místě cíle odkazů. Při přechodu do finální verze (odstraněním příkazu `\draft`) samozřejmě lejbíky zmizí.

Při tvorbě dokumentu lze využít příkazy `\rfc{<poznámka>}`, které v sazbě neudělají nic. Je-li ale zapnutý `\draft`, pak se souhrnný seznam těchto `<poznámek>` vypíše na úplně poslední stranu dokumentu a je zpětně prolinkován s místy, kde byly jednotlivé příkazy `\rfc` použity. RFC je zkratka za request for correction. Inspirace:

```

\rfc{Tady musím doplnit obrázek}
...
\rfc{Ověřit, zda hodnoty v tabulce jsou OK}

```

Jakmile je aktivován `\draft`, můžete příkazem `\linespacing=<násobek>` určit řádkování větší než implicitní řádkování 1. Například `\linespacing=1.7`. Tím se mezi řádky ve výstupním PDF dokumentu objeví mezery, do kterých může korektor v pracovní verzi dokumentu vpisovat své poznámky. Při každé změně `\linespacing` je třeba `TeX`ovat aspoň dvakrát, aby se srovnalo stránkování v obsahu.

Upozorňuji, že řádkování rozdílné od implicitního řádkování 1, je pouze pro účely pracovních verzí. Finální verze dokumentu *musí* mít řádkování 1. Ignorování této zásady bude považováno za nedodržení oficiálního stylu pro závěrečné práce na ČVUT. Proto taky CTUstyle při odstranění příkazu `\draft` automaticky deaktivuje nastavení `\linespacing`.

Velké mezerování mezi řádky bylo dříve doporučováno pro psaní studentských závěrečných prací, ale všichni lidé, kteří něco vědí o typografii, se snaží toto desítky let staré nařízení (vyplývající z technologie mechanických psacích strojů a z normy, podle které autor odevzdával své rukopisy pořízené na takovém psacím stroji tiskárně) jednoznačně vypudit jako něco, co nemá při dnešních možnostech pořizování dokumentů žádné opodstatnění. Typografie je nástroj, kterým předáváme své myšlenky dalším čtenářům a ten nástroj nesmí čtenáře rušit a unavovat ve čtení. Zmíněná starodávná norma měla za úkol usnadnit tiskárenskému závodu počítat počet znaků knihy, které autor dodal v rukopise, a na základě toho určit cenu prací. Pokud je potřeba zjistit počet znaků v současném dokumentu, můžete to udělat jednodušeji, například příkazem:

```
pdftotext dokument.pdf - | wc -m
```

Tento příkaz spočítá i znaky v automaticky generovaném obsahu. Pokud toto není žádoucí, je možné přepínačem `-f` programu `pdftotext` specifikovat, od které stránky PDF dokumentu má začít číst. Je třeba tam uvést absolutní číslo strany PDF dokumentu, nikoli čísla podle stránkových číslic.

Příkaz `\savetoner` umožní vypnout (provizorně při `\draft`) modré podklady pod listingy a tabulkami. Ve finálním dokumentu (po vypnutí `\draft`) jsou podklady vždy podbarveny.

Příkaz `\blackwhite` přepne modrou barvu na šedou. Chcete-li tisknout nakonec černobíle, je možná lepší použít tuto variantu dokumentu. Pro finálně vygenerované PDF (které není určeno k tisku) ovšem doporučuji vrátit se k barvě.

Implicitně se předpokládá tisk na duplexové tiskárně. Příkaz `\onesideprinting` přepne záhlaví do formy vhodné pro jednostranný tisk.

2.8 Teze práce

V případě disertační práce se vyžaduje vytvořit teze práce, tedy extrakt práce ve formátu A5, který mívá pro titulní a informativní stránku předepsaný formát a je omezen na cca 20 stran textu. Od verze CTUstyle May 2014 je do balíčku zahrnut makrosoubor `ctustyle-ts.tex` a výchozí příklad `example-ts.tex`, pomocí kterého snadno vytvoříte teze vaší práce. Soubor `example-ts.tex` si překopírujte do svého souboru a vyměňte připravené údaje za údaje dle vaší práce. V tomto souboru je uvedeno, které údaje jsou povinné, které nepovinné a pomocí kterých je možné přepsat implicitní texty.

Implicitní texty jsou šity na teze *disertační* práce (a jsou připraveny v češtině a angličtině). Ovšem jejich předefinováním si můžete vytvořit teze jakékoli práce.

Při psaní tezí práce je využít CTUstyle, takže vše funguje stejně, jak je posáno zde. Rozdíl je jen v tom, že není vhodné text členit na kapitoly (jen na sekce a podsekce) a pro automatické generování obsahu je potřeba přímo použít příkaz `\maketoc`. Vše je uvedeno v ukázce.

Výsledný PDF soubor s tezí práce můžete zpracovat příkazem `pdfbook soubor.pdf` a výsledek ve tvaru `soubor-book.pdf` vytisknout na duplexové tiskárně. Dostanete svazek listů A4, který se přehnutím v půli promění v knížtičku se stránkami A5. Příkaz `pdfbook` je z balíčku `pdfjam`, který se opírá o `pdflatex`. Je také možné \LaTeX obejít a udělat to přímo v `plainTeXu`, jak je popsáno v [2] v sekci 11.8.

Kapitola 3

Poznámky k typografii

Šablona CTUstyle řeší následující věci, které jsou na sobě víceméně nezávislé:

- Strukturu dokumentu a vymezení jeho povinných částí.
- Způsob, jak vyznačovat jednotlivé části ve zdrojovém textu dokumentu.
- Vzhled výstupu, neboli typografii. To je obsahem této sekce.

Vyšel jsem ze zadání v dodatku A. Dále jsem čerpal z *Grafického manuálu identity ČVUT* [9], který určuje, jak mají vypadat tiskoviny naší univerzity. V tomto manuálu je doporučeno střídat kromě černé barvu **Pantone 300 C (blankytně modrou)** jako výrazný znak tiskovin ČVUT. Tomuto doporučení jsem vyhověl. Mým cílem bylo oživit typografii závěrečných prací tak, aby se to s radostí četlo i psalo. Je v tom skryto trochu hravosti a rozpustilosti, ale domnívám se, že jen v takové míře, v jaké není narušen slavnostní ráz a důležitost studentské závěrečné práce. Navržená šablona může být použita na všech fakultách ČVUT.

Domnívám se, že modrá barva na monitoru působí dobře a při tisku na černobílé tiskárně se holt promění v šedou, ale to je pro prezentaci práce v tištěné podobě dostující. Barevný tisk je navíc pro studenty stále dostupnější.

Předpokládám, že se uživatel nebude v barvách omezovat, když bude vkládat do dokumentu schémata a obrázky, ovšem jistou střídmost by měl dodržet.

Oranžová barva (doplňková k modré) je jen navigační. Naznačuje čtenáři, že je vyznačená oblast textu klikací. Tyto oranžové rámečky zcela zmizí při tisku, protože vytištěný text už pochopitelně klikací není.

Nevyhověl jsem doporučení manuálu [9] v případě fontů, protože toto doporučení bylo v rozporu s požadavkem z dodatku A. Tam se požaduje písmo Latin Modern, zatímco v Grafickém manuálu se požaduje písmo Times. Mé oči vnímají obě písma jako hodně okoukaná. Na novém a neokoukaném písmu se podílím v jiném projektu, ale ten je nyní teprve v počátcích a není k zařazení do šablony bohužel připraven. Rozhodl jsem se zůstat u Latin Modern i z toho důvodu, že v tomto písmu vypadají hodně dobře matematické vzorečky.

Předpokládám, že dokument bude tištěn na duplexové tiskárně, po svázání listů budou liché stránky vpravo a sudé stránky vlevo. V souvislosti s tím jsou nastaveny vnitřní okraje větší, protože předpokládám, že tyto okraje se „utopí“ ve vazbě závěrečné práce. Konečně plovoucí záhlaví je navrženo tak, že svými světle modrými tečkami z vazby jakoby vychází na obě strany k vnějšímu okraji.



Literatura

- [1] Petr Olšák. *První setkání s T_EXem*, 1999, 2012.
<http://petr.olsak.net/ftp/cstex/doc/prvni.pdf>.
- [2] Petr Olšák. *T_EX pro pragmatiky*, 2013.
<http://petr.olsak.net/tpp.html>.
- [3] Petr Olšák. *T_EXbook naruby*. Konvoj Brno, 2001.
<http://petr.olsak.net/tbn.html>.
- [4] Petr Olšák. *Typografický systém T_EX*. Konvoj Brno, 1998.
<http://petr.olsak.net/tst.html>.
- [5] Donald Ervin Knuth. *Computer & Typesetting A: The T_EXbook*. Addison Wesley, 1994.
- [6] Petr Olšák. OPmac, 2012.
<http://petr.olsak.net/opmac.html>.
- [7] Vít Zýka. *FELthesis – L^AT_EX templates for thesis on CTU FEL*, 2012.
<http://zyka.net/felthesis/felthesis.zip>.
- [8] ČSN ISO 690. Informace a dokumentace – pravidla pro bibliografické odkazy a citace informačních zdrojů, duben 2011.
<http://www.citace.cz/>.
- [9] *Grafický manuál identity Českého vysokého učení technického v Praze*. Reklamní a umělecká agentura Klubko 55.
<http://www.cvut.cz/informace-pro-media/graficky-manual>.
- [10] Ludmila Tichá, Zdenka Cívínová, Michaela Morysková, Ilona Trtíková, and Lenka Němečková. *Jak psát vysokoškolské závěrečné práce*, 2012.
<http://knihovna.cvut.cz/studium/jak-psat-vskp/>.
- [11] ČVUT FEL. *Směrnice děkana pro magisterské státní závěrečné zkoušky na ČVUT FEL*, 2012.
<http://www.fel.cvut.cz/rozvoj/smerniceMSZZ.html>.
- [12] ČSN 01 6910. *Úprava písemností zpracovaných textovými editory*, duben 2007.
<http://typotypo.wz.cz/csn016910.pdf>.

Příloha A

Zadání práce

Tento dokument specifikuje šablony pro \LaTeX a MS Word, které jsou doporučeny pro psaní bakalářských, diplomových nebo disertačních prací na ČVUT FEL. Specifikace se opírá o dokumenty [10–12].

Šablony mají splňovat následující požadavky:

- Písmo Latin Modern (v \LaTeX instalacích je standardně obsaženo, pro MS Word bude OTF verze s podporou matematiky přiložená k šabloně). Velikost základního písma 11 bodů.
- Implicitní kódování šablon UTF-8.
- Formátování na papír A4, vnitřní okraj 30 mm pro pevnou vazbu, délka řádky přizpůsobena velikosti písma.
- Implicitně se předpokládá oboustranná sazba.
- Strukturní elementy: titulní list, poděkování, prohlášení, abstrakt + klíčová slova (cz/en), obsah, seznam symbolů/zkratek, přílohy, bibliografie, tabulky a obrázky s popisky.
- Číslování stránek od 1. strany textu (úvodu); úvodní stránky číslovány římsky. Důvodem je snadno rozpoznatelný rozsah práce.
- V záhlaví stránky číslo a název hlavní kapitoly. V patičce u vnějšího okraje číslo stránky.
- Součástí šablony bude styl pro bibliografie s číselnými odkazy; v seznamu literatury, řazení dle pořadí citování.
- Šablona umožní následující varianty výsledného dokumentu:
 - bakalářská/diplomová/disertační práce (předpokládá se stejná základní struktura, jen změna podtitulků),
 - anglický nebo český jazyk textu (vzory dělení, nadpisy, číslování kapitol),
 - pracovní verze (draft) s textem „Draft + datum” v patičce.

Příloha B

Zkratky a symboly

Tento text je až na výjimky převzat z [7].

B.1 Zkratky

Jako příklad pro popis zkratek poslouží pojmy ze světa \TeX .

\TeX	Program na přípravu elektronické sazby vysoké kvality vytvořený Donaldem Knuthem. Program zahrnuje interpret makrojazyka. Název programu se vyslovuje „tech“.
METAFONT	Program a makro jazyk pro generování fontů z vektorového do bitmapového formátu vytvořený Donaldem Knuthem.
METAPOST	Program generující vektorovou grafiku založený na METAFONTu vytvořený Johnem Hobby.
plain \TeX	Originální \TeX ový formát (rošíření na úrovni makrojazyka). Je součástí každé distribuce \TeX u a je vytvořen Donaldem Knuthem.
$\mathcal{C}\text{\S}plain$	\TeX ový formát rozšiřující plain \TeX o možnosti sazby v českém a slovenském jazyce vytvořený Petrem Olšákem.
\LaTeX	Nejznámější \TeX ový formát (rozšíření na úrovni makrojazyka) vytvořený Leslie Lamportem. Existuje obiludné množství různých balíčků, které pomocí makrojazyka \TeX u dále rozšiřují výchozí možnosti \LaTeX u. Rozličné uživatelské požadavky jsou nejčastěji řešeny použitím vhodného balíčku.
OPmac	Olšákovy Plain \TeX ová makra nabízející uživatelům plain \TeX u podobné možnosti, jako \LaTeX , ovšem přímočařeji a jednodušeji.
Con \TeX t	Typografický systém vystavěný na rozšíření \TeX u s názvem Lua \TeX (kombinuje makrojazyk \TeX u s jazykem Lua) a na množství předpřipravených makro souborů vytvořený týmem v čele s Hansem Hagenem. Rozličné uživatelské požadavky jsou nastavovány pomocí přiřazení hodnot klíčovým slovům společně s možností \TeX ového, METAPOSTího a Lua programování.

B.2 Symboly

- π Konečná verze \TeX u zmíněna v Knuthově \TeX tamentu.
- e Konečná verze METAFONTu.
- 2 ε Současná verze \LaTeX u používaná od roku 1994. Počítá se s ní jako s přechodnou verzí mezi původní Lamportovou verzí \LaTeX 2.09 a cílovou verzí \LaTeX 3. Tento přechodný stav už trvá 19 let.