



FAKULTA APLIKOVANÝCH VĚD
ZÁPADOČESKÉ UNIVERZITY
V PLZNI

KATEDRA INFORMATIKY
A VÝPOČETNÍ TECHNIKY

Diplomová práce

FASThesis – návod k použití šablony závěrečné práce na FAV ZČU

Kamil Ekštein



FAKULTA APLIKOVANÝCH VĚD
ZÁPADOČESKÉ UNIVERZITY
V PLZNI

KATEDRA INFORMATIKY
A VÝPOČETNÍ TECHNIKY

Diplomová práce

FASThesis – návod k použití šablony závěrečné práce na FAV ZČU

Ing. Kamil Ekštein, Ph.D.

Vedoucí práce

Prof. Ing. Tomáš Marný, Ph.D.

© Kamil Ekštejn, 2024.

Všechna práva vyhrazena. Žádná část tohoto dokumentu nesmí být reprodukována ani rozšiřována jakoukoli formou, elektronicky či mechanicky, fotokopírováním, nahráváním nebo jiným způsobem, nebo uložena v systému pro ukládání a vyhledávání informací bez písemného souhlasu držitelů autorských práv.

Citace v seznamu literatury:

EKŠTEJN, Kamil. *FASThesis – návod k použití šablony závěrečné práce na FAV ZČU*. Plzeň, 2024. Diplomová práce. Západočeská univerzita v Plzni, Fakulta aplikovaných věd, Katedra informatiky a výpočetní techniky. Vedoucí práce Prof. Ing. Tomáš Marný, Ph.D.

ZÁPADOČESKÁ UNIVERZITA V PLZNI

Fakulta aplikovaných věd

Akademický rok: 2021/2022

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(projektu, uměleckého díla, uměleckého výkonu)

Jméno a příjmení:

Osobní číslo:

Studijní program: **B3902 Inženýrská informatika**

Studijní obor: **Informatika**

Téma práce:

Zadávací katedra: **Katedra informatiky a výpočetní techniky**

Zásady pro vypracování

1. Seznamte se se stávajícími ; zdokumentujte jejich výhody a nevýhody.
2. Seznamte se se stávající podobou
3. Na základě bodů 1 a 2 navrhnete
4. Navržený implementujte a zabezpečte z bodu 2.
5. Výsledné řešení důkladně otestujte na netriviální množině dat
6. Vytvořte komunitně srozumitelný a veřejně dostupný tutoriál.
7. Zhodnoťte dosažené výsledky.

Rozsah bakalářské práce: **doporuč. 30 s. původního textu**
Rozsah grafických prací: **dle potřeby**
Forma zpracování bakalářské práce: **tištěná/elektronická**

Seznam doporučené literatury:

Dodá vedoucí bakalářské práce.

Vedoucí bakalářské práce: **Doc. Ing. Ph.D.**
Katedra informatiky a výpočetní techniky

Datum zadání bakalářské práce: **24. srpna 2021**
Termín odevzdání bakalářské práce: **5. května 2022**

L.S.

Doc. Ing. Miloš Železný, Ph.D.
děkan

Doc. Ing. Přemysl Brada, MSc., Ph.D.
vedoucí katedry

Prohlášení

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně a výhradně s použitím citovaných pramenů, literatury a dalších odborných zdrojů. Tato práce nebyla využita k získání jiného nebo stejného akademického titulu.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona v platném znění, zejména skutečnost, že Západočeská univerzita v Plzni má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona.

V Nové Vsi u Nového Města na Moravě dne 14. února 2024

.....

Kamil Ekštejn

V textu jsou použity názvy produktů, technologií, služeb, aplikací, společností apod., které mohou být ochrannými známkami nebo registrovanými ochrannými známkami příslušných vlastníků.

Abstrakt

Text představuje a blíže popisuje třídu (class file) \LaTeX ového dokumentu `fastthesis`, která se používá k sazbě kvalifikačních prací na Fakultě aplikovaných věd Západočeské univerzity v Plzni. Vysvětluje, jak správně používat šablonu kvalifikační práce `fastthesis` pro sazecí systém \LaTeX a jak podobu výsledného dokumentu ovlivňují různé přepínače a příkazy, kterými se sazba při použití této šablony řídí.

Kromě toho, že tento abstrakt skutečně stručně shrnuje obsah dokumentu, tak také slouží jako ukázka, jak by měl abstrakt vypadat. Délka abstraktu by se měla pohybovat mezi 100 a 300 slovy, a ačkoliv není dáno žádné konkrétní omezení, určitě by neměla překročit polovinu stránky A4. Abstrakt má čtenáři poskytnout představu o tom, co se v dokumentu dočte (a tedy jestli vůbec má smysl se do čtení pouštět).

Abstract

The text presents and describes the \LaTeX document class `fastthesis`, which is used for the typesetting of theses at the Faculty of Applied Sciences, University of West Bohemia in Pilsen. It explains how to properly use the `fastthesis` template for \LaTeX typesetting system and how the resulting document is affected by various switches and commands that control the typesetting process when using this template.

Not only this abstract really briefly summarises the document's contents, it also serves as an example of what an abstract might look like. The length of the abstract should be between 100 and 300 words, and although no specific limit is given, it should certainly not exceed half an A4 page. The abstract is intended to give the reader a rough idea of what they will read in the document (and therefore whether it is even worth getting started with it at all).

Klíčová slova

šablona kvalifikační práce • sazba • DTP • \LaTeX

Poděkování

Na tomto místě bych rád poděkoval svému předchůdci v roli „local wizarďa“, člověku, který připravil první prakticky použitelnou L^AT_EXovou šablonu kvalifikační práce na Katedře informatiky a výpočetní techniky Fakulty aplikovaných věd Západočeské univerzity v Plzni, Ing. Petru Lobazovi, Ph.D. Asi ještě větší a srdečnější poděkování si zaslouží úžasný člověk, matematik, který mě s T_EXem před téměř třiceti lety seznámil: Doc. RNDr. František Ježek, CSc.

Rád bych na tomto místě poděkoval i dalšímu matematikovi, Doc. Ing. Janu Pospíšilovi, Ph.D., který mne neúnavně přesvědčoval, že FAV novou šablonu kvalifikačních prací potřebuje jako sůl a následně mě nutil na ní pracovat (a samozřejmě přispěl řadou podnětných připomínek).

Díky si zaslouží i mí drazí kolegové z Katedry informatiky a výpočetní techniky, kteří svojí konstruktivní kritikou významně přispěli ke konečné podobě šablony – jmenovitě jsou to tito (v abecedním pořadí): Doc. Ing. Přemysl Brada, M.Sc., Ph.D., Doc. Ing. Pavel Herout, Ph.D., Ing. Martin Úbl a Ing. Petr Vaněček, Ph.D.

Zvláštní poděkování pak patří Ing. Františku Pártlovi, který šablonu intenzivně testoval, pomáhal opravovat chyby, neúnavně se staral o gitový repozitář šablony (což bylo s ohledem na můj vztah ke gitu zvláště náročné) a zejména zajišťoval integraci šablony do Overleafu.

V neposlední řadě si poděkování zaslouží také studenti, kteří aktivně pomáhali při testování nově vzniklé šablony: Bc. Veronika Báčová a Petra Ocelíková.

Kamil Ekštejn,
autor šablony
(únor 2024)

Note for English readers

In case that your command of the Czech language is not sufficient for a perfect understanding of the whole of this document, please be so kind and open the file `manual_en.pdf` which contains the same text (i.e. the user's manual for the \LaTeX template `fastthesis.cls`) in English.

Obsah

1 Úvod	5
1.1 Instalace	5
2 Použití šablony	7
2.1 Nastavení třídy fastthesis	7
2.1.1 Seznam přepínačů třídy fastthesis	8
2.2 Minimální kostra dokumentu třídy fastthesis	10
2.3 Některé úvodní „stavové“ příkazy	11
2.3.1 Nastavení jména autora práce	11
2.3.2 Nastavení identifikačního čísla práce v informačním systému ZČU STAG	11
2.3.3 Vložení zadání kvalifikační práce	12
2.3.4 Specifikace souboru s bibliografickou databází	12
2.3.5 Abstrakt	13
2.3.6 Klíčová slova	14
2.3.7 Nastavení označení druhu práce	15
2.4 Sazba titulních stránek	16
2.5 Sazba poslední stránky (desek/přebalu)	17
3 Sazba některých specifických částí textu	20
3.1 Matematika	20
3.1.1 Vzorce, rovnice a jejich soustavy	20
3.2 Obrázky, ilustrace, grafy, diagramy	21
3.3 Tabulky	24
3.4 Zdrojový kód	26
3.5 Terminál/konzole operačního systému	27
3.6 Když je něco opravdu hodně důležité	30
3.7 Užitečné drobnosti	31
A Lokální instalace fontu GT America	34

B	Přehled balíčků importovaných třídou fastthesis	35
C	Často kladené dotazy (FAQ)	38
	Bibliografie	40
	Seznam obrázků	41
	Seznam tabulek	42
	Seznam výpisů	43

Předmluva

Milý autore/milá autorko,

tímto se do Vašich schopných rukou dostává **šablona kvalifikační práce** na Fakultě aplikovaných věd Západočeské univerzity v Plzni. Tato šablona je určena k dosažení jednotného vzhledu kvalifikačních prací sázených v typografickém systému L^AT_EX. Při jejím návrhu jsem se řídil zejména tím, aby bylo její používání pro (často nezkušeného) autora textu co nejsnazší, ale aby zároveň výsledný produkt odpovídal zásadám a pravidlům moderní sazby technických dokumentů, působil dostatečně reprezentativně a zároveň lehce a přirozeně, a přitom byl přehledný a dobře čitelný. Grafická podoba šablony pak reflektuje pravidla stanovená **Manuálem jednotného vizuálního stylu** Západočeské univerzity v Plzni¹.

Při tvorbě šablony jsem vycházel z obdobných šablon, které jsou autorům kvalifikačních prací k dispozici na významných českých i zahraničních vysokých školách. Snažil jsem se je důkladně analyzovat, abych „to dobré“ začlenil i do naší šablony, a naopak „tomu špatnému“ abych se vyhnul. Grafické a typografické prvky, vyskytující se v těchto šablonách, jsem hodnotil na základě poznatků čerpaných ze tří skvělých knih: *Graphic Design Now* [FF05] od Charlotte a Petera Fiellových, *Jazyk grafického designu* [Pou12] od Richarda Poulina a *Typografický manuál* [Ber+16] od Vladimíra Berana.

Výše popsaným způsobem šablonu ovlivnily šablony kvalifikačních prací *Kalifornského technologického institutu* (Caltech), *Harvardovy univerzity*, *Princetonské univerzity*, *Newyorské univerzity*, *Univerzity v Oslu*, *Vysokého učení technického v Brně*, *Matematicko-fyzikální fakulty Univerzity Karlovy*, *Fakulty informatiky Masarykovy univerzity* a řada dalších. Důležitou roli pochopitelně sehrála i naše předchozí L^AT_EX-ová šablona, připravená před řadou let Petrem Lobazem.

Tato šablona by měla řešit prakticky všechny zásadní aspekty sazby kvalifikační práce (a pokud je přímo neřeší, vždy je možné je řešit ve vlastní režii použitím příkazů T_EXu), nicméně – protože je zatím (v únoru 2024) docela nová a „nezajatá“ – může se stát, že v ní odhalíte chybu, nedostatek, nějaké nekonzistentní chování

¹Z tohoto manuálu vychází kromě podoby logotypů univerzity a jejích součástí zejména barevnost a použití konkrétních fontů.

apod. Pokud se tak stane, budu rád, když mi napíšete, o co jde a jak se problém projevuje, e-mailem na adresu `kekstein@kiv.zcu.cz` a já se pokusím zajistit co možná nejdříve nápravu. Někdy se ovšem může jednat o situaci, kterou programátoři popisují obvykle (v anglickém originále) větou: „It’s not a bug, it’s a feature.“

Doufám, že se Vám s touto šablonou bude pracovat dobře a pomůže Vám vytvořit perfektní kvalifikační práci, která bude následně úspěšně obhájena. K tomu Vám přeju z pozice jejího tvůrce hodně štěstí a držím palce.

Tato *šablona kvalifikační práce* nastavuje převážnou většinu parametrů sazby v typografickém systému \LaTeX tak, aby výsledný dokument odpovídal jednak požadavkům norem (pokud existují) a jednak tradičním zvyklostem úpravy vědeckotechnických dokumentů. Je navržena tak, aby nebylo nutné žádné parametry sazby „ručně“ upravovat. Zkušenosti uživatelé \LaTeX u nicméně pochopitelně mohou vzhled sazby snadno změnit použitím příkazů \LaTeX u a maker, která poskytuje základní třída dokumentu využitá v této šabloně. Zejména z tohoto důvodu – pro zkušené uživatele – je tedy dobré vědět, že

třída `fastthesis` této šablony je založena na třídě `memoir`.

Lze tedy používat kompletní aparát třídy `memoir` [Wil22], je-li to vhodné (ovšem není to vůbec nutné).

Šablona ovlivňuje zásadním způsobem vzhled výsledného dokumentu tak, aby jednak jasně deklaroval příslušnost k Fakultě aplikovaných věd Západočeské univerzity v Plzni¹, druhá aby byl text přehledný a dobře čitelný a aby bylo pokud možno „o vše postaráno“ automaticky a autor nemusel řešit např. vzhled a provedení citací použité literatury, umístění a přesné znění prohlášení atp.

1.1 Instalace

Šablonu není nutné na váš počítač vůbec instalovat (stejně jako celý \TeX) v případě, že hodláte k sepsání kvalifikační práce využít online editor **Overleaf** – na adrese URL <https://www.overleaf.com/read/ryhpnsmtgrrs> je šablona pro tento editor již k dispozici, připravena k okamžitému použití. Pro studenty FAV ZČU se

¹Šablona je k dispozici jako open source, a je tedy možné ji libovolně modifikovat za podmínky dodržení ustanovení licence LGPL. Pokud tedy nejste z FAV ZČU a chcete tuto šablonu použít, směřte do toho: Bude zřejmě třeba změnit barevnost a zejména **font!** Proč? Použitý bezplatkový font nadpisů *GT America* není volně k dispozici – ZČU je vlastníkem komerční licence na jeho použití, neboť je nedílnou součástí jejího vizuálního stylu. Jste-li tedy z jiné instituce, která nemá tento font zakoupený, použít jej nemůžete a je **nutné** jej nahradit jiným (doporučuji třeba volně dostupný font *Roboto Condensed*).

pak nabízí ještě lepší možnost, a to využít lokální instalace Overleafu/Share \LaTeX u na serveru Katedry informatiky a výpočetní techniky FAV ZČU na adrese URL <https://overleaf.kiv.zcu.cz> – tam se totiž neuplatňuje omezení na čas kompilace dokumentu jako v případě zdarma dostupného Overleafu.

Pokud ovšem používáte \TeX pravidelně a máte jej tedy nainstalovaný na svém počítači (což je silně **doporučená** varianta), není instalace šablony také nijak komplikovaná. Postupujte podle krok za krokem podle níže uvedeného návodu:

1. Stáhněte si instalační archiv `fastthesis.zip`.
2. **Vytvořte** pro projekt (tedy vaši kvalifikační práci) **samostatnou složku/adresář**.
3. Do vytvořené složky **nakopírujte** ze staženého instalačního archivu **soubor** `fastthesis.cls` – to je základní a nejdůležitější soubor, kterým je šablona tvořena – definice třídy dokumentu `fastthesis`.
4. Dále do vytvořené složky **zkopírujte jako podsložku složku** `img` – v té se nacházejí obrázky, které šablona využívá (zejména loga fakulty a kateder a tématické grafické motivy na titulní stránku).
5. Do složky pak **také uložte zadání vaší kvalifikační práce ve formátu PDF**, lhostejno, zda je naskenované nebo vygenerované informačním systémem univerzity. Na názvu tohoto PDF souboru nezáleží, do dokumentu je zadání vloženo příkazem `\assignment{zadani.pdf}`, jehož parametrem je právě tento název (může být i s cestou do nějaké podsložky).
6. Nakonec je třeba nainstalovat font *GT America*, který je předepsán **Manuálem jednotného vizuálního stylu** Západočeské univerzity v Plzni jako font nadpisů. Tuto netriviální akci popisuje detailně příloha A.

Pokud by informace z přílohy A nebyly dostatečně úplné a instalace fontů by vám činila problémy, využijte stránky organizace TUG (= *TeX Users' Group*), které se tématu instalace fontů věnují – najdete je na URL <https://tug.org/fonts/fontinstall.html>.

Použití šablony

2

Šablonu tvoří (kromě jiného) zejména soubor definice třídy \LaTeX ového dokumentu, tzv. *classfile*, nazvaný `fastthesis.cls`, který \LaTeX u poskytuje třídu dokumentu `fastthesis`.

Zdrojový text vaší práce tedy je obyčejný textový soubor (v kódování UTF-8¹), jehož úvodním příkazem je `\documentclass{fastthesis}`. Jeho detailní nastavení prostřednictvím nepovinných parametrů bude popsáno dále.

2.1 Nastavení třídy `fastthesis`

Třída dokumentu `fastthesis` má celou řadu „přepínačů“, tedy nepovinných parametrů, které ovlivňují, jak bude výsledný dokument vypadat. Základní nastavení odpovídá použití příkazu v této podobě:

```
\documentclass[czech, kiv, ba, he, iso690alph, pdf]{fastthesis}
```

Výsledný dokument tedy bude vysázen v českém jazyce (“`czech`”), bude použito logo a označení Katedry informatiky a výpočetní techniky (“`kiv`”), nadpis na titulní straně (a jinde) uvede práci jako bakalářskou kvalifikační práci (“`bc`”), jejím autorem je osoba používající pro sebe osobní zájmeno „on“ (“`he`”), bibliografické informace budou formátovány podle normy ISO ČSN 690 (“`iso690alph`”) v podobě doporučené např. webem <https://www.citace.com> nebo <https://citace.zcu.cz>, přičemž zdroje budou v textu identifikovány zkratkou jména autora (či prvními písmeny jmen autorů, je-li jich více) a posledními dvěma číslicemi roku vydání publikace (např. [AB2023]). Finální produkt sazby bude používán zejména online v podobě dokumentu ve formátu PDF (“`pdf`”) nebo bude vytištěn na kvalitní barevné tiskárně – nepředpokládá se tedy primárně tisk dokumentu na černobílé tiskárně.

¹Je sice možné celkem snadno šablonu „překódovat“ do jiného kódování znaků národních abeced, ale při současném stavu užívání různých operačních systémů, editorů a dalších podpůrných nástrojů se to nejvíce jako rozumné.

2.1.1 Seznam přepínačů třídy fastthesis

Tabulka 2.1 na str. 8 přehledně shrnuje všechny použitelné přepínače, které ovlivňují chování šablony. Některé z nich se (pochopitelně) navzájem vylučují (např. “he” a “she”) a jsou-li užity současně, buď se provede pouze poslední z nich, nebo se dokument dokonce může začít chovat podivně (neměl by, ale kdo ví).

Tabulka 2.1: Seznam všech použitelných přepínačů třídy dokumentu fastthesis

přepínač	vysvětlení
czech	Dokument bude sázen v českém jazyce, budou použity české verze log a názvů univerzity a jejích součástí, v češtině budou všechny automaticky vkládané texty (prohlášení apod.), české bude dělení slov v dokumentu atp.
english	Dokument bude sázen v anglickém jazyce, budou použity anglické verze log a názvů univerzity a jejích součástí, v angličtině budou všechny automaticky vkládané texty (prohlášení apod.), anglické bude dělení slov v dokumentu atp.
he	Text píše osoba používající pro sebe osobní zájmeno „on“.
she	Text píše osoba používající pro sebe osobní zájmeno „ona“.
sem	Dokument je seminární práce.
ba	Dokument je bakalářská kvalifikační práce.
bc	Dokument je bakalářská kvalifikační práce (<i>alias přepínače “ba”</i>).
ma	Dokument je diplomová, tj. magisterská kvalifikační práce.
mgr	Dokument je diplomová, tj. magisterská kvalifikační práce (<i>alias přepínače “ma”</i>).
ing	Dokument je diplomová, tj. magisterská kvalifikační práce (<i>alias přepínače “ma”</i>).
ddt	Dokument je rigorózní práce (= <i>doctoral dissertation theses</i>).
rig	Dokument je rigorózní práce (<i>alias přepínače “ddt”</i>).
phd	Dokument je doktorská disertační práce.
hab	Dokument je habilitační práce.
oth	Dokument je něco jiného (= <i>other</i>) a z výše uvedených možností si nelze vybrat. V tomto případě je pak nutné dodatečně nastavit některé parametry šablony použitím k tomu určených příkazů, viz sekce 2.3.7 na str. 15.

(tabulka pokračuje na další stránce)

Tabulka 2.1 (pokračování z předchozí stránky)

přepínač	vysvětlení
kgm	Práce je odevzdávána na Katedře geomatiky.
kfy	Práce je odevzdávána na Katedře fyziky.
kiv	Práce je odevzdávána na Katedře informatiky a výpočetní techniky.
kma	Práce je odevzdávána na Katedře matematiky.
kme	Práce je odevzdávána na Katedře mechaniky.
kky	Práce je odevzdávána na Katedře kybernetiky.
ftalph	Formátování bibliografických informací je provedeno podle interních pravidel šablony (ve shodě s normou ČSN ISO 690).
iso690numb	Formátování bibliografických informací je provedeno tak, jak doporučuje norma ČSN ISO 690 a některé organizace či iniciativy, zabývající se etikou vědeckého výzkumu a specificky správným citováním (např. weby https://www.citace.com či „náš“ https://citace.zcu.cz), přičemž jednotlivé zdroje jsou pak označeny vzestupně čísly, např. takto: [1], [2], [3].
iso690alph	Shodné s „iso690numb“, ale jednotlivé zdroje jsou označeny zkratkou jména autora či prvních písmen jmen autorů a posledními dvěma číslicemi roku vydání publikace, např. takto: [Kra19], [ČM23].
iso690auyr	Shodné s „iso690numb“, ale jednotlivé zdroje jsou označeny jmény autorů a datem vydání publikace, např. takto: (Král, 2019), (Čáp; Mák, 2023).
pdf	Dokument je určen primárně k uložení a prohlížení online ve formátu PDF, je tedy vysázen barevně a s grafickými motivy.
prn	Dokument je určen primárně k tisku na černobílé tiskárně, tzn. nebude vysázen s barevnými (typo)grafickými motivy a grafikou na pozadí titulní stránky.
nobggfx	Dokument bude vysázen barevně, ale bez grafického motivu na pozadí titulní stránky (= <i>no background graphics</i>).

(tabulka pokračuje na další stránce)

Tabulka 2.1 (pokračování z předchozí stránky)

přepínač	vysvětlení
viewonly	Dokument je určen pouze pro prohlížení na obrazovce počítače – nebude tištěn. Neobsahuje proto prázdné sudé stránky na jednostranně potištěných listech (pro pokročilé T _E Xisty: odpovídá přepínači “oneside” standardních L ^A T _E Xových tříd book, report atp., a samozřejmě také použité základní třídy memoir).

2.2 Minimální kostra dokumentu třídy fastthesis

L^AT_EXový kód níže ukazuje základní, nejjednodušší použití třídy fastthesis pro sazbu dokumentu. Některé příkazy *není nutné* použít, resp. nezpůsobí to chybu při zpracování dokumentu L^AT_EXem, ale v takovém případě se vysázejí kódem šablony přednastavené hodnoty, což by ve skutečné kvalifikační práci mohlo být docela na závadu.

```

1 \documentclass[czech, ing, kiv, he]{fastthesis}
2 \title{\texttt{FASTthesis} — návod k~použití šablony zá...}
3 \author{Kamil}{Ekštein}{Ing.}{Ph.D.}
4 \supervisor{Prof. Ing. Tomáš Marný, Ph.D.}
5 \stagworkid{12345}
6 \assignment{zadani.pdf}
7 \signdate{31}{12}{2022}{V Nové Vsi u~Nového Města na Moravě}
8 \addbibresource{manual.bib}
9 \abstract{Text abstraktu v~jazyce práce, tj. zde česky.}
10 {The abstract text in a secondary language, here in English.}
11 \keywords{šablona kvalifikační práce, sazba, DTP, \LaTeX}
12 \acknowledgement{Text poděkování.}
13 \begin{document}
14 \frontpages[tm]
15 \tableofcontents
16 \chapter{Úvod}
17 ...
18 \appendix
19 \chapter{První příloha}
20 ...
21 \backmatter
22 \printbibliography
23 \backpage
24 \end{document}

```

Úvodní („stavové“) příkazy před příkazem `\begin{document}` je samozřejmě možné uvést v libovolném pořadí, protože sazba textu podle jimi nastavených hodnot se provede najednou až vydáním příkazu `\frontpages`.

2.3 Některé úvodní „stavové“ příkazy

Většina úvodních příkazů, nastavujících zejména texty v tzv. *front matteru* (jako je např. titul, patitul, různá prohlášení apod.), je „samovysvětlující“ – podívejte se do zdrojového kódu tohoto manuálu (který je součástí distribučního balíčku) a mělo by být jasno. Dále tedy budou uvedeny pouze ty příkazy, které v sobě skrývají nějaké záludnosti.

2.3.1 Nastavení jména autora práce

Zatímco u běžně používaných L^AT_EXových tříd (*article*, *report*, *book* apod.) i u základní třídy *memoir* má příkaz `\author{jméno autora}` pouze jediný argument, kterým je jméno autora (míněno celé jméno, včetně křestních a dalších jmen a případných titulů) v takové podobě, v jaké si ho autor přeje vysázet na titulní stránku, u této šablony (tedy třídy *fastthesis*) je situace složitější:

Z řady technických důvodů je nutné, aby šablona „znala“ i jednotlivé komponenty jména. Proto má modifikovaný příkaz `\author` v dokumentu třídy *fastthesis* podobu `\author{křestní jméno}{příjmení}{tituly před jménem}{tituly za jménem}` – tedy celkem 4 **povinné** argumenty.

Ač třeba v okamžiku, kdy píšete např. diplomovou práci, ještě nemáte titul(y) za jménem, není možné některý z argumentů vynechat. Všechny 4 argumenty musí být uvedeny, ale některé mohou být samozřejmě prázdné:

```
\author{Jan}{Novák}{Bc.}{}
```

2.3.2 Nastavení identifikačního čísla práce v informačním systému ZČU STAG

V informačním systému STAG, který používá ZČU (a řada jiných univerzit), jsou všechny kvalifikační práce (nejen odevzdané a obhájené, ale i rozpracované) uloženy pod *unikátním číselným identifikátorem*, který dovoluje práci zobrazit (je-li správně použit v konkrétním URL) a případně i stáhnout ve formátu PDF.

Tento unikátní identifikátor práce je možné šabloně předat příkazem `\stag-workid{identifikátor práce}`, načež šablona zajistí jeho zakódování do QR kódu, který pak (pokud si to autor přeje) lze umístit na zadní stranu desek práce jako součást grafického motivu (detailně popsáno v sekci 2.5 na straně 17). Užití ukazuje příklad:

```

1 \documentclass[czech, ma, kiv, he, pdf]{fastthesis}
2 \title{\texttt{FASTthesis} — návod k~použití šablony ...}
3 \author{Kamil}{Ekštein}{Ing.}{Ph.D.}
4 \supervisor{Prof. Ing. Tomáš Marný, Ph.D.}
5 \stagworkid{12345}
6 \assignment{zadani.pdf}
7 \signdate{14}{02}{2024}{V Nové Vsi u~Nového Města na Moravě}
8 \begin{document}
9 \frontpages[tm]
10 ...
11 (TEXT PRÁCE)
12 ...
13 \backmatter
14 \setbackpageqrqr
15 \backpage
16 \end{document}

```

Jak ve STAGu nalézt předmětný unikátní identifikátor práce je popsáno poznámce pod čarou 3 na straně 18.

2.3.3 Vložení zadání kvalifikační práce

Zadání kvalifikační práce (tak jak je uloženo ve STAGu) se do dokumentu vkládá příkazem `\assignment{jméno souboru}`. Toto zadání je třeba nejprve ze STAGu **vyexportovat ve formátu PDF** a uložit do složky, ve které je zdrojový kód dokumentu kvalifikační práce (v případě práce v Overleafu pak do kořenové složky projektu, „vedle“ samotného \TeX ového zdrojového kódu dokumentu).

Soubor PDF se zadáním může být uložen i v nějaké podsložce. Je-li tomu tak, musí být uvedena celá relativní cesta k tomuto souboru (i v případě Windows uvádějte v cestě **lomítka normální**, nikoliv zpětná), tj. např.:

```
\assignment{vloz/dokumenty/zadani.pdf}
```

Upozornění: Ze specifikovaného souboru ve formátu PDF se vloží do výsledného dokumentu *nejvýše první dvě stránky* (což by mělo stačit a mělo by to být správně, pokud je v PDF skutečně zadání kvalifikační práce – to je na FAV totiž vždy jedno- nebo dvojstránkové).

2.3.4 Specifikace souboru s bibliografickou databází

Bibliografické informace v dokumentu třídy `fastthesis` spravuje kombinace nástroje `biber` a balíku `biblatex`. Zkušenější uživatelé \TeX u se mohou ptát, proč šablona opouští léty prověřený nástroj `bibtex` s možností ovlivnit vzhled sazby bibliografických informací prostřednictvím souboru `.bst` (= *bibliography style*). Je to proto, že `bibtex` nepracuje korektně se znaky národních abeced kódovanými

v rozšířených kódováních UTF-8 apod. Backend biber je o dost novější, takže s tím problémy nemá.

Mechanismus biber + biblatex vyžaduje, aby již na začátku dokumentu byl specifikován soubor, ve kterém se nachází bibliografická databáze. To se provádí příkazem `\addbibresource{soubor s bibliografickou databází}`, takže třeba takto:

```
\addbibresource{moje_literatura.bib}
```

Obsah souboru .bib – tedy bibliografická databáze sázeného dokumentu – je stejný, jako tomu bylo v případě bibtexu. Používá se stejná syntax a totožné formátování, tzn. pokud již autor má z dřívější vlastní bibliografickou databázi ve formátu bibtex, kterou si např. budoval v průběhu studia při psaní různých prací, může ji bez úprav okamžitě použít.

Ukázka níže zobrazuje typickou podobu úseku souboru .bib s jedním konkrétním bibliografickým záznamem typu *kniha*:

```
1 @book{Poulin2012 ,
2   title="Jazyk grafického designu",
3   author="Richard Poulin",
4   publisher="Slovart",
5   year="2012",
6   address="Praha",
7   isbn="978-80-7391-552-0"
8 }
```

Na takovýto záznam se pak v textu odkazuje pomocí příkazů \TeX u `\cite{identifikátor zdroje}`:

```
1 ... najdeme např. ve vynikající knize Jazyk grafického
2 designu~\cite{Poulin2012}, která ...
```

K dalším specifickým způsobům citací zdrojů lze pak využít příkazy `\citep{...}`, `\citet{...}` atp. z balíku natbib; detaily jsou popsány např. na adrese URL https://en.wikibooks.org/wiki/LaTeX/Bibliography_Management.

Zásadní výhodou použitého formátu bibtex je fakt, že je de facto standardem v oblasti přírodovědných a vědecko-technických publikací, a tedy drtivá většina online služeb pro vyhledávání zdrojů informací (jako je např. **Google Scholar**, **IEEE Xplore**, **Web of Science**, **Scopus** atd.) umí citační záznamy v tomto formátu přímo generovat.

Úplný popis formátu bibtex pro vedení bibliografických databází pak může autor kvalifikační práce najít na webu věnovaném nástroji bibtex na adrese URL <https://www.bibtex.com/g/bibtex-format/>.

2.3.5 Abstrakt

K sazbě abstraktu práce slouží příkaz `\abstract{textu abstraktu v jazyce práce}{text abstraktu ve zvoleném cizím jazyce}`. Píše-li tedy autor práci česky, bude v prvním

argumentu příkazu celý text abstraktu v češtině, ve druhém argumentu pak celý text abstraktu přeložený do cizího jazyka, v našem oboru to patrně bude téměř vždy angličtina. Píše-li však autor kvalifikační práci **anglicky**, bude prvním argumentem **anglicky psaný abstrakt** práce a ve druhém argumentu bude odpovídající text **česky** (tady už se možnost volby nenabízí, protože kvalifikační práce je odevzdávána v Česku, na české vysoké škole). Výjimku z tohoto pravidla může mít pouze cizinec píšící kvalifikační práci anglicky, který pak pro text druhého argumentu použije svůj rodný jazyk (není-li jím angličtina).

Ve druhé popsané variantě příběhu s abstraktem by ovšem došlo k nepříjemné situaci, kdy bude text abstraktu v rodném jazyce cizince nadepsán česky (protože příkaz `\abstract` za normálních okolností používá pro nadpisy abstraktů pouze český výraz „Abstrakt“ a jeho anglický ekvivalent „Abstract“). Právě proto, aby šla tato nežádoucí eventualita vyloučit, lze použít příkaz `\abstract` s nepovinným parametrem, kterým je správná podoba nadpisu cizojazyčného abstraktu. Ukázka níže předvádí, jak by vypadal např. abstrakt práce psané anglicky autorem, jehož rodnou řečí je francouzština:

```
1 \abstract[Résumé]{This thesis describes the effect of ...}
2 {Cette thèse décrit l'effet de ...}
```

2.3.6 Klíčová slova

Klíčová slova se do dokumentu vkládají příkazem `\keywords{seznam klíčových slov oddělených čárkami}`. Klíčových slov by nemělo být moc, optimální počet je obvykle někde mezi 3 a 8 (ale není to dogma, počet se může samozřejmě lišit např. podle složitosti a rozsahu pojednávané problematiky).

Píše-li autor práci česky, uvádí se jednotlivá klíčová slova (i slovní spojení) **s malými počátečními písmeny** (všech slov ve spojení, pokud to nejsou vlastní jména či názvy produktů apod.). Naopak v angličtině se obvykle tzv. kapitalizují, tedy počáteční písmena slov (prvních slov ve slovních spojeních) se píšou velká.

V případě práce psané v češtině, a tedy českých klíčových slov to může vypadat např. takto:

```
1 \keywords{počítačové vidění, strojové učení, strategie,
2 plánování, CAM}
```

Anglická verze výše uvedeného příkladu pak správně vypadá takto:

```
1 \keywords{Computer vision, Machine learning, Strategy,
2 Planning, CAM}
```

Zkratky (technologií, produktů atp.) je možné do klíčových slov uvádět, ale autor by v takovém případě měl mít jistotu, že (i) je zkratka skutečně všeobecně známá a že (ii) nemůže kolidovat se stejnou zkratkou pro něco jiného; protože pak by si čtenář

práce mohl třeba myslet, že se dozví něco o CAM (= *Computer-Aided Manufacturing*), ale přitom práce pojednává o CAM (= *Cardiac Arrest Management*).

Hrozí-li byť i jen malá pravděpodobnost záměny, je třeba termín rozepsat a nespolehat na zkratku.

2.3.7 Nastavení označení druhu práce

Označení druhu práce na titulní straně (tj. např. “Technická zpráva”, “Uživatelská příručka” apod.) je třeba nastavovat **pouze** v případě, že jste v úvodním příkazu dokumentu `\documentclass` použili přepínač “oth” (= *other*), tedy nastavili šablonu na druh práce “jiná”. **Píšete-li bakalářku (“ba”) nebo diplomku (“ma”), vůbec tuto sekci nečtěte**, aby vás to neponoukalo...

Je-li šablona nastavena přepínačem inicializačního příkazu `\documentclass` na druh práce “jiná”, tedy např.:

```
\documentclass[czech, oth, kiv, he, pdf]{fastthesis}
```

je nutné šablonu informovat, jaký text označující druh práce se má vysázet na titulní stranu (a také např. do ukázky správné citace práce). To se provede příkazem `\worktypespec{označení druhu práce}` – zapsaný musí být ve zdrojovém textu samozřejmě před příkazem `\frontpages`, protože ten již potřebuje mít k dispozici informaci o tom, jak má být práce nadepsána.

Úvodní sekvence příkazů v případě sazby práce „jiného“ (rozumějte nikoliv standardního) druhu pak může vypadat např. takto:

```
1 \documentclass[czech, oth, kiv, he, pdf, viewonly]{fastthesis}
2 \title{Šablona závěrečné práce na FAV ZČU}
3 \worktypespec{Návod k~použití}
4 \author{Kamil}{Ekštein}{Ing.}{Ph.D.}
5 \begin{document}
6 ...
```

2.3.7.1 Dodatečný obsah úvodních stránek (tzv. *front matteru*)

Za normálních okolností (při psaní bakalářské nebo diplomové práce) šablona zajišťuje správné provedení sazby všech potřebných informací v tzv. **front matteru**, čili na úvodních stránkách dokumentu, které jsou číslovány jinou řadou než hlavní text (zde římskými číslicemi provedenými minuskami v kulatých závorkách). Do front matteru patří titul a patitül, různá prohlášení (copyright, prohlášení o autorství, prohlášení o ochranných známkách atp.), zadání práce, abstrakt, klíčová slova, poděkování atd. Jejich přesnou podobu víceméně definuje druh vytvářené práce (tedy jestli je to např. diplomka).

Pokud ale autor zvolí v inicializačním příkazu `\documentclass` přepínačem “oth” druh práce “jiná”, výše uvedené komponenty front matteru se sázet vůbec nebudou, protože není zřejmé, zda do tohoto druhu práce patří. Ovšem autor může potřebovat do front matteru zasáhnout a něco do něj umístit. K tomu účelu dává šablona k dispozici příkaz `\auxfrontmattercontent{dodatečný obsah front matteru}`. Jeho použití ilustruje následující příklad:

```

1 \documentclass[czech, oth, kiv, he, pdf, viewonly]{fastthesis}
2 \title{Šablona závěrečné práce na FAV ZČU}
3 \worktypespec{Návod k~použití}
4 \author{Kamil}{Ekštein}{Ing.}{Ph.D.}
5 \auxfrontmattercontent{
6   \section*{Poděkování}
7   Autor děkuje všem, kteří ho vytrvale podporovali v~jeho
8   úporné dřině a nedopustili, aby se zbláznil\dots
9 }
10 \begin{document}
11 ...

```

2.4 Sazba titulních stránek

Poté, co autor vyplnil všechny důležité informace týkající se jeho práce zadáním úvodních stavových příkazů s příslušnými argumenty a zahájil sazbu dokumentu příkazem `\begin{document}`, může (spíš musí) přikázat hromadnou sazbu všech titulních stránek (tj. titulu, patitulu, stránky s prohlášením, abstrakty atd.).

To se provede příkazem `\frontpages`, který musí následovat bezprostředně za zahajovacím příkazem sazby dokumentu `\begin{document}`. Tento příkaz má jeden nepovinný argument, který může nabývat hodnot “tm” nebo “notm” (s významem *trademarks* a *no trademarks*).

Pokud tedy autor v dokumentu používá názvy a označení, které jsou nebo mohou být ochrannými známkami nebo registrovanými ochrannými známkami, mělo by na začátku dokument stát **prohlášení o této skutečnosti**, které autora ochrání před případnými právními následky užívání ochranných známek či registrovaných ochranných známek v textu bez uvedení, že jde o ochranné známky nebo registrované ochranné známky nějakých konkrétních firem či společností.

Jakmile se tedy v práci píše o Windows, MATLABu, Oracle, AutoCADu atp., je nutné použít příkaz v této podobě:

```
\frontpages[tm]
```

To zajistí, že za patitulem v dolní části stránky bude automaticky vysázeno vhodné prohlášení o ochranných známkách, které autora práce zproští případné právní odpovědnosti za neoprávněné užití ochranné známky nebo registrované ochranné známky.

V případě tvorby bakalářské, diplomové či disertační práce je to prakticky nutnost. U semestrální práce je to naopak přehnané a nejspíš zbytečné. Pak tedy lze použít příkaz v následující podobě:

```
\frontpages[notm]
```

Při použití příkazu `\frontpages` bez argumentu se sází varianta s prohlášením (tedy jako by byl uveden nepovinný argument ve tvaru “tm” – tedy předvolená je „bezpečnější“ varianta).

2.5 Sazba poslední stránky (desek/přebalu)

Třída `fastthesis` umožňuje autorovi dokumentu jednoduše vysadit také poslední stránku, tj. tu, která je na společném archu s titulní stranou, a tvoří tak zadní stranu desek (nebo přebalu, chcete-li) dokumentu. Udělá se to příkazem `\backpage`, který nemá žádné parametry.

Grafická úprava zadní strany desek je shodná s úpravou titulní stránky, tj. použije se stejný grafický motiv příslušné katedry na pozadí (pokud ovšem není v úvodním příkazu `\documentclass` použit přepínač “nobggfx”, tedy „nesázet grafiku na pozadí“) a pruh v heraldické barvě Fakulty aplikovaných věd o stejné šířce a ve stejné výšce, jako na titulce (tzn. pruhy na sebe navazují, resp. je to jediný pruh přes celý arch, na kterém je přední i zadní strana desek).

Výše uvedené se ovšem stane pouze tehdy, je-li dokument sázen jako plnobarevný, tzn. v úvodním příkazu `\documentclass` byl použit přepínač “pdf”² (nebo nebyl použit přepínač “prn”).

Doprostřed grafického motivu poslední stránky je v případě přání autora možné umístit buď malý obrázek (v libovolném grafickém formátu, přednostně samozřejmě vektorovém) nebo **QR kód**.

Jediným smyslem umístění obrázku na zadní stranu desek je případné zvýšení estetické úrovně dokumentu (ale to se nemusí vždy povést a estetická úroveň může takto být i snížena): Má-li autor k dispozici např. nějaký grafický symbol, zajímavý vizuální motiv či něco podobného, co charakterizuje vytvořený dokument a chce si jej takto „označit“ a tím odlišit od ostatních kvalifikačních prací, může tak učinit pomocí příkazu `\setbackpagepic{jméno_souboru}`, který má jediný parametr, a tím je jméno (případně včetně celé cesty) souboru s vkládaným grafickým obsahem. Jméno souboru v příkazu `\setbackpagepic` může být uvedeno bez přípony, je-li uložen v některém z běžných grafických formátů:

```
\setbackpagepic{img/sierpinski_triangle}
```

²Tento přepínač nemusí být explicitně uveden, protože je ve zdrojovém kódu třídy `fastthesis` nastavený jako předvolený.

Smysl vložení QR kódu je mnohem hlubší a zásadnější. Každá kvalifikační práce (přínejmenším na školách, které to myslují aspoň trochu vážně) je totiž uložena v nějakém informačním systému, kde ji lze nalézt a případně prohlédnout nebo i stáhnout ve formátu PDF. Do QR kódu na zadní straně desek je tedy možné umístit tu adresu URL, na které je dokument uložen, a tak vytvořit trvalé a jednoznačné propojení tištěné a elektronické verze předmětné kvalifikační práce. To je samozřejmě mnohem důležitější a praktičtější než sebezajímavější obrázek, a proto je **silně doporučeno** na zadní stranu sázet právě tento QR kód.

QR kód se na zadní stranu desek vysází příkazem `\setbackpageqrcode`, který lze použít buď zcela bez parametru nebo s jedním parametrem, a jeho použití je maličko složitější (a souvisí s příkazem `\stagworkid`, zmíněným v části 2.3.2 na straně 11):

1. V případě, že kvalifikační práci píšete a odevzdáváte u nás na FAV, stačí příkaz `\setbackpageqrcode` zapsat bez parametru, pokud jste předtím v hlavičce dokumentu nastavili příkazem `\stagworkid` jednoznačné identifikační číslo práce, pod kterým je vedena v informačním systému STAG³.
2. V případě, že situace popsaná v bodě 1 není aplikovatelná, je možné použít příkaz `\setbackpageqrcode` s jedním parametrem, jehož obsahem je URL, určující, kde se práce nachází. Např. takto:

```

1 ...
2 \setbackpageqrcode{https://doc.abc.cz/jan.novy/DP.pdf}
3 \backpage
4 \end{document}

```

3. Poslední možnost se týká situace, kdy práce má nějaký unikátní identifikátor podobný výše zmíněnému identifikátoru v informačním systému STAG, ale informační systém je jiný a tudíž URL začínají zcela jiným prefixem, definujícím umístění daného informačního systému (tato varianta se tedy týká především autorů, kteří chtějí šablonu využívat, ale nejsou z prostředí ZČU). V takovém případě se nejprve použije příkaz `\setqrcodebaseurl`, jehož jediným parametrem je bazová adresa, k níž bude následně připojen unikátní identifikátor práce, a takto vzniklý URL se teprve zakóduje do QR kódu.

Příkazy `\setbackpagepic` a `\setbackpageqrcode` (případně i `\setqrcodebaseurl`) je **nutné napsat ještě před použitím příkazu `\backpage`**:

³To najdete ve STAGu: Studium → Prohlížení → Kvalifikační práce, vyplníte např. své osobní číslo a stisknete **Hledat**. V nalezených záznamech (je-li jich více) najdete svoji práci, kliknete na ni a pak vpravo za textem "Tisk/export" vyhledáte tlačítko se šipkou doprava (vypadá to jako symboly pro pauzu a přehrávání na audiotechnice). Toto tlačítko vygeneruje URL pro přímý přístup k příslušné kvalifikační práci. Z URL potřebujete jen tu poslední část, což je právě unikátní identifikátor práce. Je to to číslo, které v URL následuje za sekvencí "praceIdno=".

```
1 ...
2 \setbackpagepic{img/sierpinski_triangle}
3 \backpage
4 \end{document}
```

Je to proto, že zmíněné příkazy nic přímo na stránku nesázejí, ale jen ukládají do vnitřních proměnných šablony informace, které posléze využije příkaz `\backpage` pro sazbu zadní strany desek.

Varianta s nastavením bazového URL podle bodu 3 vypadá takto:

```
1 ...
2 \setqrcodebaseurl{https://www.mujserver.cz/prace/show?wrkid=}
3 \setbackpageqrcode
4 \backpage
5 \end{document}
```

V případě, že byl v hlavičce dokumentu uveden příkaz `\stagworkid{123456}`, bude v QR kódu na zadní straně desek zakódované URL `https://www.mujserver.cz/prace/show?wrkid=123456`, tzn. části, předané příkazům `\setqrcodebaseurl` a `\stagworkid` se prostě jednoduše zřetězí.

Sazba některých specifických částí textu

3

3.1 Matematika

3.1.1 Vzorce, rovnice a jejich soustavy

3.1.1.1 Sazba vzorců

Vzorec v řádce. Jednotlivé izolované vzorce je možné sázet do řádky souvislého textu použitím tzv. „jednodolarového prostředí“, tedy např. `\sin(x)=0`. Tento postup je ovšem vhodný pouze pro krátké, nekomplikované vzorečky, na které **není třeba se odkazovat z jiných částí textu**.

Vzorec na středu řádky, opticky oddělený. K sazbě složitějších, byť stále izolovaných a z textu neodkazovaných vzorců slouží tzv. „dvoj dolarové prostředí“, které zajistí vysazení vzorečku na střed stránky, přičemž nad a pod vzorcem je vynechaný řádek, čímž je vzorec dostatečně opticky oddělen od ostatního souvislého textu. Sazba do „dvoj dolarového prostředí“ se provádí např. takto:

```
1 $$\frac{\mathrm{d}f}{\mathrm{d}t} = \lim_{h \rightarrow 0} \frac{f(t+h) - f(t)}{h}$$
2
```

Výsledek pak vypadá takto:

$$\frac{df}{dt} = \lim_{h \rightarrow 0} \frac{f(t+h) - f(t)}{h}$$

Nicméně ani tento způsob není úplně vhodný pro sazbu rozsáhlejších matematicky orientovaných textů právě proto, že nedovoluje se na vzorec odkazovat z textu – vzorec nemá totiž žádný identifikátor.

Vzorec v prostředí equation. Pro sazbu vzorců má \LaTeX připravené prostředí `equation`, které vzorec sází na střed stránky a automaticky zajistí jeho jednoznač-

nou identifikaci uvedením čísla ve formátu “(číslo kapitoly.pořadí vzorce)”:

$$i\hbar \frac{\partial}{\partial t} \Psi = H\Psi \quad (3.1)$$

Toto je nejelegantnější z uvedených řešení, neboť na vzorec lze snadno odkazovat, např. takto: „viz předpis Schrödingerovy rovnice (3.1) na str. 21...“, a zároveň je opticky vhodně oddělen od průběžného textu. Zdrojový kód výše uvedené ukázky vypadá takto:

```
1 \begin{equation}
2 i\hslash\frac{\partial}{\partial t}\Psi = H\Psi
3 \label{eqn:schroedinger}
4 \end{equation}
```

3.1.1.2 Sazba rovnic a jejich soustav

Při sazbě rovnic je patrně nejdůležitějším estetickým požadavkem umístění znaménka rovnosti všech rovnic na stejné vertikále, díky čemuž vypadá soustava dobře opticky „srovnaná“. Toto zajišťuje prostředí eqnarray:

$$\nabla \cdot \mathbf{E} = \frac{\rho}{\epsilon_0} \quad (3.2)$$

$$\nabla \cdot \mathbf{B} = 0 \quad (3.3)$$

$$\nabla \times \mathbf{E} = -\frac{\partial \mathbf{B}}{\partial t} \quad (3.4)$$

$$\nabla \times \mathbf{B} = \mu_0 \left(\mathbf{J} + \epsilon_0 \frac{\partial \mathbf{E}}{\partial t} \right) \quad (3.5)$$

Uvedením příkazu `\label` na vhodném místě (před příkazem řádkového zlomu `\\`, kterým se ukončují jednotlivé rovnice soustavy) lze odkazovat i na samostatné jednotlivé rovnice, např. takto: „... jak je vidět z podoby Gaussova zákona v diferenciálním tvaru (3.2) na str. 21...“. Zdrojový kód ukázky následuje níže:

```
1 \begin{eqnarray}
2 \nabla\cdot\mathbf{E} & = & \frac{\rho}{\epsilon_0}
3 \label{eqn:gaussLaw}\\
4 \nabla\cdot\mathbf{B} & = & 0\\
5 \nabla\times\mathbf{E} & = & -\frac{\partial\mathbf{B}}{\partial t}\\
6 \nabla\times\mathbf{B} & = & \mu_0\left(\mathbf{J} + \right. \\
7 & & \left. \epsilon_0\frac{\partial\mathbf{E}}{\partial t}\right)
8 \end{eqnarray}
```

3.2 Obrázky, ilustrace, grafy, diagramy

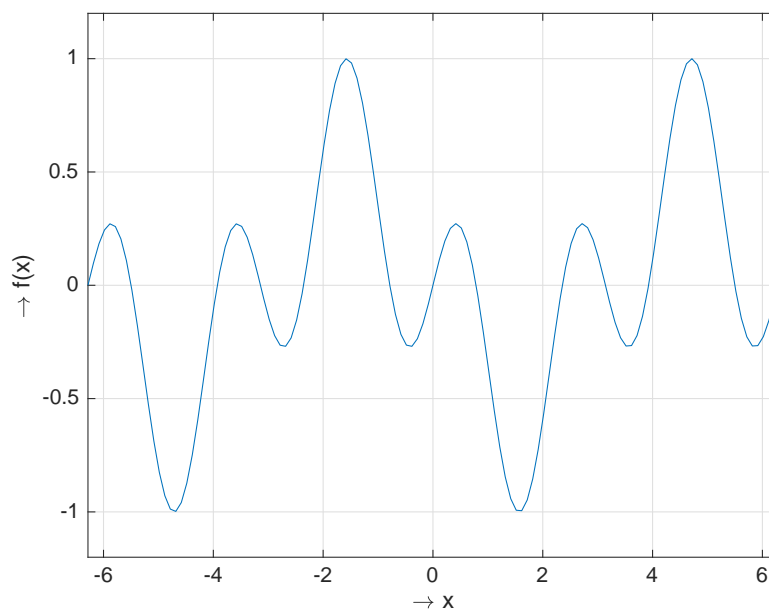
Obrázky, ilustrace, grafy, diagramy a podobné objekty se typicky sází do tzv. **plovoucích prostředí**, které – jak už název napovídá – plovou volně v textu a mohou

proplouvat celkem bez omezení i mezi stránkami. Autoři dokumentů často svádí s \TeX em zoufalý a marný boj, snažíce se ho přimět, aby daný obrázek vysázel přesně na dané konkrétní místo, např. mezi dvě věty textu. To se téměř nikdy nestane a byť \TeX nabízí určité postupy, jak vynutit vysazení obrázku na určité místo, není to vlastně ani žádoucí.

\TeX dokáže o optimálním umístění plovoucího prostředí rozhodnout sám na základě rozvržení ostatních typografických prvků a parametrů sazby tak, aby stránku příliš „nerozbil“. Nejrozumnější je mu v tom nebránit. Ovšem – protože se může stát, že se např. obrázek nebo tabulka ocitne docela daleko od textu, ke kterému se vztahuje – každý plovoucí objekt **musí být jednoznačně identifikován** číslem a nejlépe i stručným popisem. Pro vkládání grafických souborů je ve třídě `fastthesis` k dispozici aparát balíčku `graphics`, zejména „mocný“ příkaz `\includegraphics`. Vložení např. grafu funkce, vycentrovaného na stránce, o šířce 75 % šířky řádku se provede tímto kódem:

```
1 \begin{figure}[ht]
2 \centering
3 \includegraphics[width=.75\textwidth]{img/sincos}
4 \caption{Průběh funkce  $f(x) = \sin(x)\cos(2x)$ }
5 \label{fig:funcSinCos}
6 \end{figure}
```

Výsledek provedení tohoto kódu je vidět na obr. 3.1 na str. 22. Patrně nejdůležitější



Obrázek 3.1: Průběh funkce $f(x) = \sin(x)\cos(2x)$

věc při vkládání obrázků do dokumentů sázených \TeX em je jejich formát.

Vždy, když je to jen trochu možné, dáváme přednost použití grafiky ve **vektorovém formátu**. Rastrovou (bitmapovou^a) grafiku užijeme pouze tehdy, kdy z principu není možné daný grafický motiv uložit ve vektorovém formátu, tedy např. v případě fotografií či snímků obrazovky. Neexistuje žádný přijatelný důvod pro vkládání rastrové grafiky v případě grafů funkcí, vývojových či jiných diagramů, schémat, plánů atp.

^aTa ale nemusí být nutně uložena jen v souboru s příponou .bmp, ale také .jpg, .png atd.

Je třeba si uvědomit, že tisk (i v případě starých a nepříliš kvalitních tiskáren) probíhá s rozlišením minimálně 300 DPI, spíš více. Při tisku v rozlišení 300 DPI tak rastrová grafika, která má být vytištěná na stránce na šířku textové řádky (tedy asi 13 cm), musí mít horizontálně alespoň 1535 pixelů. Má-li méně, a \TeX je i přesto přinucen grafiku roztáhnout na požadovanou velikost, dojde k viditelnému „rozkostičkování“. U vektorové grafiky pochopitelně k tomuto jevu nedochází – vektorovou grafiku lze zvětšovat/zmenšovat zcela libovolně bez vlivu na kvalitu.

Doporučené volně dostupné nástroje pro přípravu ilustrací/grafů/diagramů ve vektorovém formátu jsou např.:

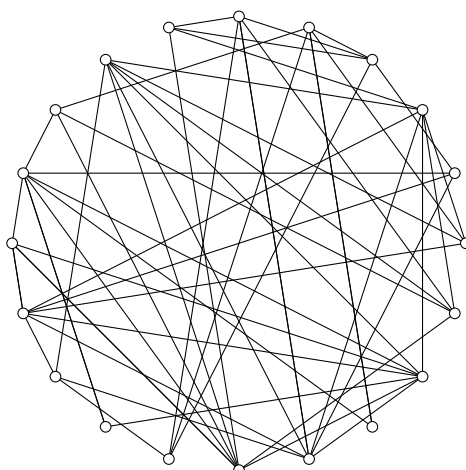
- (i) **yEd** – <https://www.yworks.com/products/yed>
- (ii) **Inkscape** – <https://inkscape.org>
- (iii) **Graphviz** – <https://graphviz.org>
- (iv) **Diagrams.net** – <https://app.diagrams.net>

Třída `fastthesis` zavádí (pro vlastní potřebu) balíček `tikz`, který dovoluje vektorovou grafiku vytvářet přímo ve zdrojovém kódu dokumentu pomocí speciálního jazyka geometrických primitiv. Autor dokumentu tedy může tento výkonný nástroj v případě potřeby okamžitě využít. Níže uvedený kód Shenga Bau [Bau22] vysadí náhodně vygenerovaný graf zobrazený na obr. 3.2 na str. 24:

```

1 \begin{tikzpicture}
2 \foreach \i in {0, 1, ..., 19}
3 {
4   \foreach \j in {1, 2, 3}
5   {
6     \pgfmathsetmacro{\thenum}{int(random(0, 19))}
7     \draw (18 * \i:3) — (18 * \thenum:3);
8   }
9 }
10 \foreach \i in {0, 1, ..., 19}
11 {
12   \filldraw [fill=white,draw=black] (18 * \i:3) circle(2pt);
13 }
14 \end{tikzpicture}

```

Obrázek 3.2: Náhodný graf vysazený TikZem podle kódu Shenga Bau [Bau22]

Poslední **důležitá věc, týkající se sazby obrázků**, ilustrací, grafů apod. je popis těchto objektů: Tradičně se umísťuje **pod** příslušný grafický objekt a obvykle se sází na střed stránky (z tohoto pravidla lze v odůvodněných případech činit výjimky). Popisek obrázku začíná velkým písmenem a neukončuje se tečkou. Pokud se popisek skládá z více vět, pak jsou ukončeny tečkou, nicméně poslední z nich opět být tečkou ukončena nemusí (je to situace analogická s nadpisy).

3.3 Tabulky

Třída `fastthesis` při své inicializaci zavádí dva užitečné \TeX ové balíky `booktabs` a `longtable`, které modifikují a doplňují základní možnosti sazby tabulek, nabízené \LaTeX em¹.

Při psaní kvalifikační práce se hodí zejména jedna vlastnost tabulek z balíku `longtable`, a to ta, že tyto tabulky mohou pokračovat přes několik stránek a pomocí příkazů z balíku lze dokonce specifikovat, jak bude vypadat záhlaví/zápatí nejen v případě začátku a konce tabulky, ale i v případě jejího rozdělení na více stránek – záhlaví/zápatí může být obecně jiné v případě začátku/konce tabulky a v případě jejího rozdělení (tj. je např. možné vysázet pod poslední řádek tabulky na jedné straně třeba text „tabulka pokračuje na další straně“, viz tabulka 2.1 na straně 8 tohoto dokumentu).

Jednoduchou tabulku (s předpřipravenou možností zlomu přes několik stránek) lze vysázet za pomoci příkazů z balíku `longtable` např. takto:

¹ Podívejte se na jejich dokumentace na webu CTAN (Comprehensive \TeX Archive Network) na URL <https://ctan.org/pkg/longtable>, resp. <https://ctan.org/pkg/booktabs>, kde jsou detailně popsány všechny možnosti jejich použití, včetně těch velmi speciálních.

```

1 \begin{center}
2 \begin{longtable}{rr}
3 \caption{Přehled jednotek množství informace}
4 \label{tab:infunits}\\
5 \toprule[1.5pt]
6 \textbf{jednotka} & \textbf{množství informace}\\
7 \midrule
8 \endfirsthead
9 \multicolumn{2}{c}{\tablename{}~\thetable{}}
10 \textit{(pokračování z~předchozí stránky)}}\\
11 \midrule
12 \textbf{jednotka} & \textbf{množství informace}\\
13 \midrule
14 \endhead
15 \midrule
16 \multicolumn{2}{r}{%
17 \textit{(tabulka pokračuje na další stránce)}}\\
18 \endfoot
19 \bottomrule[1.5pt]
20 \endlastfoot
21 1 KiB & 1024 B\\
22 1 MiB & 1048576 B\\
23 1 GiB & 1073741824 B\\
24 1 TiB & 1099511627776 B\\
25 1 PiB & 1125899906842624 B\\
26 \end{longtable}
27 \end{center}

```

Zpracováním výše uvedeného zdrojového kódu vysází \LaTeX tabulku v této podobě:

Tabulka 3.1: Přehled jednotek množství informace

jednotka	množství informace
1 KiB	1024 B
1 MiB	1 048 576 B
1 GiB	1 073 741 824 B
1 TiB	1 099 511 627 776 B
1 PiB	1 125 899 906 842 624 B

Pochopitelně, že tahle tabulka rozdělená přes více stránek není (protože je na to moc krátká), ale ze zdrojového kódu je patrné, jak pomocí příkazů `\endfirsthead`, `\endhead`, `\endfoot` a `\endlastfoot` „vysvětlit“ \TeX u, jak vypadá první a všechna

další záhlaví tabulky, resp. poslední zápatí a všechna předchozí před tímto posledním. Tabulka se tedy může lámat přes více stránek: Na začátku je použita definice záhlaví končící příkazem `\endfirsthead`, na další stránce pak bude tabulka začínat opět záhlavím, ale tentokrát podle definice ukončené příkazem `\endhead`. Analogicky to funguje pro zápatí. Detaily na URL <https://ctan.org/pkg/longtable>.

Povšimněte si, prosím, že u tabulek (a také třeba u výpisů zdrojového kódu) se ve shodě s tradičními pravidly sazby **umísťuje popisek nad příslušnou tabulkou**.

3.4 Zdrojový kód

Třída dokumentu `fastthesis` využívá pro sazbu zdrojového kódu balík `listings`, který se automaticky připojí, jakmile šablonu použijete. Je tedy možné v textu přímo používat příkazy a prostředí, které balík `listings` poskytuje, zejména tedy prostředí `lstlisting`, přičemž vzhled výsledného výpisu kódu se nastavuje pomocí příkazů `\lstdefinestyle{mujstyl}{...}` a `\lstset{style=mujstyl}`.

Šablona dává uživateli k dispozici styl výpisu zdrojového kódu pojmenovaný “`FASTthesisLstStyle`” (vzhled odpovídá tomu v ukázce zdrojového kódu 3.1 na str. 27). Autor tedy může tento předpřipravený styl využít a ručně pak třeba dodefinovat další požadované vlastnosti výpisu zdrojového kódu, např. takto:

```
1 \lstset{style=FASTthesisLstStyle, numberblanklines=false,
2       tabsize=5, keywordstyle=\color{red}}
3 \begin{lstlisting}
4 ...
5 \end{lstlisting}
```

Řídíme-li si sazbu zdrojového kódu vlastními silami, je třeba jen mít na paměti, že výpis **musí být opatřen popiskem** s číslem výpisu – tento popisek se sází nad samotným výpisem (tak jako je to na ukázce 3.1). Je to proto, aby bylo možné se na kód z textu práce odkazovat. Ze stejného důvodu **musí být řádky výpisu číslované**. Odkazování z textu pak vypadá např. takto: „...typické použití popsané konstrukce lze vidět na ř. 2 zdrojového kódu 3.1 na str. 27.“

Pro méně zdatné uživatele \LaTeX u je v definici třídy pomocí balíku `listings` připraveno prostředí `code`, které se používá např. takto:

```
1 \begin{code}{C}{Ukázkový výpis kódu v~jazyce ANSI C}
2 int fact(int n) {
3   if (n <= 0)
4     return 1; /* protože 0! = 1 */
5   else
6     return n * fact(n - 1);
7 }
8 \end{code}
```

Vysázený výpis zdrojového kódu v dokumentu pak vypadá takto:

Zdrojový kód 3.1: Ukázkový výpis kódu v jazyce ANSI C

```
1 int fact(int n) {
2     if (n <= 0)
3         return 1; /* proto že 0! = 1 */
4     else
5         return n * fact(n - 1);
6 }
```

Pokud se z textu na zdrojový kód odkazuje, je nutné umístit **za** popis cílové návěští odkazu příkazem `\label{identifikátor návěští}`. V případě výpisu zdrojového kódu se identifikátoru návěští obvykle předřazuje prefix “lst”:

```
1 \begin{code}{C}{Výpočet faktoriálu v~C\label{lst:faktorial}}
2 ...
3 \end{code}
```

Seznam všech výpisů zdrojových kódů (a také všech výpisů obsahu konzole, viz část 3.5) se vygeneruje příkazem `\listoflistings`, který patří do části dokumentu zvané **back matter**, tedy na konec práce, za bibliografii, spolu s případným seznamem obrázků, tabulek, vzorců, indexem atp.

3.5 Terminál/konzole operačního systému

Máloco vypadá v sazbě kvalifikačních a podobných prací (a obecně technických dokumentů) tak špatně, jako nekvalitní screenshot **konzole** – nebo chcete-li terminálu či příkazového řádku – operačního systému (totéž platí pochopitelně pro terminálový přístup k serveru).

Prakticky všechny operační systémy hlavního proudu používají (patrně z historických důvodů) v terminálech černou (nebo jinou velmi tmavou barvu) pro pozadí a bílou (nebo jinou světlou) barvu pro text. Toto v kombinaci s výrazně nižší hustotou bodů monitorů² než mají běžné laserové tiskárny, a také s faktem, že řada terminálových aplikací stále používá „staré“ rastrové fonty, vede většinou k nepříjemnému výsledku: Na půlce stránky je černý obdélník (tj. obrovská a zcela zbytečná spotřeba toneru), na kterém jsou nepřírodně velká, „zubatá“, rozkostičkováná písmena, často ještě v důsledku uložení do formátu JPEG notně poškozená artefakty ztrátové komprese.

Třída dokument *fastthesis* proto nabízí autorům prací prostředí *console*, které dovoluje vysázet výpisy v terminálu/konzoli operačního systému „kulturně“.

²Monitor má typicky rozlišení od 72 PPI v případě starých kusů s malým množstvím pixelů (z dnešního pohledu) po zhruba 200 u špičkových moderních tzv. HiDPI monitorů grafických pracovních stanic. Ovšem tisk na laserové tiskárně (i docela staré) začíná na 300 DPI...

Zda půjde o příkazový řádek operačního systému Windows nebo o terminál unixového systému, se rozlišuje použitím dvou různých tzv. promptů – `\winprompt` a `\uxprompt`.

Ve zdrojovém kódu dokumentu může vypadat použití prostředí console ve variantě s windowsovským promptem např. takto:

```

1 \setwinprompt{C:/Windows/System32}
2 \begin{console}{Ukázkový výpis obsahu příkazového
3   řádku Windows\label{lst:demoConWin}}
4 '\winprompt 'ipconfig
5
6 Windows IP Configuration
7
8
9 Ethernet adapter Ethernet:
10
11     Connection-specific DNS Suffix  . : xyz.net
12     Link-local IPv6 Address . . . . . : bc95::1255:cd70:71...
13     IPv4 Address. . . . . : 192.168.1.101
14     Subnet Mask . . . . . : 255.255.255.0
15     Default Gateway . . . . . : 192.168.1.1
16
17 Ethernet adapter Ethernet 2:
18
19     Connection-specific DNS Suffix  . :
20     Link-local IPv6 Address . . . . . : aa71::c21d:2cf8:25...
21     IPv4 Address. . . . . : 192.168.56.1
22     Subnet Mask . . . . . : 255.255.255.0
23     Default Gateway . . . . . :
24
25 '\winprompt '

```

Příkaz `\setwinprompt{aktivní složka}` nastavuje aktivní složku, ve které se uživatel právě nachází a kde zadává příkazy, a která je tak součástí promptu. Při použití tohoto příkazu se aktivní složka (nebo chcete-li *cesta* čili *path*) zapisuje s normálními lomítky jako oddělovači složek, nikoliv se zpětnými. Je to proto, že zpětné lomítko uvozuje příkazy \TeX u. Každopádně vysázené to bude správně se zpětnými lomítky, tak jak to v operačním systému Windows má být – příkaz `\setwinprompt` si normální lomítka sám převede na zpětná.

Vysázení promptu zajistí příkaz `\winprompt`, který by ovšem \TeX ignoroval (protože je uvnitř prostředí pro doslovný výpis), takže je nutné ho tzv. *escapovat*, tedy v tomto konkrétním případě uzavřít do zpětných apostrofů, což \TeX u říká: „Tento příkaz nesázet, ten provést!“

Vysázený výpis příkazu a jeho výstupu z ukázky výše v příkazovém řádku operačního systému Windows pak vypadá takto:

Výpis 3.2: Ukázkový výpis obsahu příkazového řádku Windows

```

1 C:\Windows\System32>ipconfig
2
3 Windows IP Configuration
4
5
6 Ethernet adapter Ethernet:
7
8     Connection-specific DNS Suffix  . : xyz.net
9     Link-local IPv6 Address . . . . . :
10    bc95::1255:cd70:711b:223%11
11    IPv4 Address. . . . . : 192.168.1.101
12    Subnet Mask . . . . . : 255.255.255.0
13    Default Gateway . . . . . : 192.168.1.1
14
15 Ethernet adapter Ethernet 2:
16
17     Connection-specific DNS Suffix  . :
18     Link-local IPv6 Address . . . . . :
19    aa71::c21d:2cf8:252f:8a35%7
20    IPv4 Address. . . . . : 192.168.56.1
21    Subnet Mask . . . . . : 255.255.255.0
22    Default Gateway . . . . . :
23
24 C:\Windows\System32>

```

Obdobně funguje sazba terminálu operačních systému unixového typu. K nastavení pracovního adresáře³ se v případě unixového terminálu použije obdobný příkaz `\setuxprompt{uživatelské jméno vč. stroje}{pracovní adresář}`, který má ovšem argumenty 2: (i) název stroje, na kterém uživatel pracuje, kterému obvykle předchází i uživatelské jméno daného uživatele, a (ii) právě používaný, aktivní pracovní adresář, zapisovaný běžným způsobem užívaným v UNIXu:

```

1 \setuxprompt{Kamil@CML}{/usr/bin}
2 \begin{console}{Ukázkový výpis obsahu terminálu Linuxu
3   \label{lst:demoConUx}}
4   '\uxprompt{ls -la | grep "^.{48}\a"}
5   -rwxr-xr-x  1 Kamil None      53956 Nov 13 13:03 addftinfo.exe
6   -rwxr-xr-x  1 Kamil None       3075 Feb 11 20:45 addgnupghome
7   -rwxr-xr-x  1 Kamil None    1150349 Jan 21 12:58 addr2line.exe
8   -rwxr-xr-x  1 Kamil None    165785 Nov 13 13:03 afmtodit
9   -rwxr-xr-x  1 Kamil None     15433 Dec 22 10:05 afslog.exe
10  -rwxr-xr-x  1 Kamil None     50220 Oct 22 20:23 agetty.exe
11  -rwxr-xr-x  1 Kamil None      2217 Feb 11 20:45 applygnupg...

```

³Teď se jedná o operační systém unixového typu, a je tedy na místě použít terminologii UNIXu. Ostatně „místo“, kde se uživatel v hierarchické struktuře souborového systému právě nachází, vypisuje příkaz „`pwd`“, tedy *print working directory*.

```

12 -rwxr-xr-x 2 Kamil None 1177671 Jan 21 12:58 ar.exe
13 -rwxr-xr-x 1 Kamil None 35876 Nov 15 18:07 arch.exe
14 -rwxr-xr-x 2 Kamil None 1837207 Jan 21 12:58 as.exe
15 -rwxr-xr-x 1 Kamil None 108312 Jan 24 20:50 ash.exe
16 -rwxr-xr-x 1 Kamil None 27040 May 14 2022 autopoint
17 -rwxr-xr-x 1 Kamil None 686902 Dec 16 02:40 awk.exe
18 '\uxprompt '
19 \end{console}

```

Následkem provedení výše uvedeného zdrojového kódu dostaneme toto:

Výpis 3.3: Ukázkový výpis obsahu terminálu Linuxu

```

1 Kamil@CML:/usr/bin$ ls -la | grep "^.{48\}a"
2 -rwxr-xr-x 1 Kamil None 53956 Nov 13 13:03 addftinfo.exe
3 -rwxr-xr-x 1 Kamil None 3075 Feb 11 20:45 addgnupghome
4 -rwxr-xr-x 1 Kamil None 1150349 Jan 21 12:58 addr2line.exe
5 -rwxr-xr-x 1 Kamil None 165785 Nov 13 13:03 afmtodit
6 -rwxr-xr-x 1 Kamil None 15433 Dec 22 10:05 afslog.exe
7 -rwxr-xr-x 1 Kamil None 50220 Oct 22 20:23 agetty.exe
8 -rwxr-xr-x 1 Kamil None 2217 Feb 11 20:45
   applygnupgdefaults
9 -rwxr-xr-x 2 Kamil None 1177671 Jan 21 12:58 ar.exe
10 -rwxr-xr-x 1 Kamil None 35876 Nov 15 18:07 arch.exe
11 -rwxr-xr-x 2 Kamil None 1837207 Jan 21 12:58 as.exe
12 -rwxr-xr-x 1 Kamil None 108312 Jan 24 20:50 ash.exe
13 -rwxr-xr-x 1 Kamil None 27040 May 14 2022 autopoint
14 -rwxr-xr-x 1 Kamil None 686902 Dec 16 02:40 awk.exe
15 Kamil@CML:/usr/bin$

```

3.6 Když je něco opravdu hodně důležité

Třída dokumentu `fastthesis` nabízí autorům kvalifikačních prací zvláštní prostředí pojmenované `important`, jehož úkolem je *silně graficky akcentovat* určitou část textu, kterou autor považuje za mimořádně důležitou. Ovšem **pozor, není dobré toto prostředí nadužívat**, protože opticky silně narušuje souvislý tok textu a také nutnost často zvýrazňovat určité pasáže jako mimořádně důležité může znamenat, že autor nemá koncepci textu řádně promyšlenou.

Jak se prostředí `important` při sazbě dokumentu používá, ukazuje následující úsek zdrojového kódu:

```

1 \begin{important}
2 Tento text považuje autor práce z~nějakého důvodu za
3 mimořádně důležitý a je za každou cenu nutné, aby si ho
4 čtenář povšiml a věnoval mu maximální možnou pozornost.
5 Toho je dosaženo zmenšením šířky sazby a ve zbylém místě
6 na okrajích je text výrazně orámován červenou (nebo černou)

```

```

7 v~případě nastavení třídy dokumentu \filename"fastthesis"
8 na černobílou sazbu přepínačem ‘‘\command"prn"’’) barvou.
9 \end{important}

```

Výsledek vidíte v následujícím odstavci:

Tento text považuje autor práce z nějakého důvodu za mimořádně důležitý a je za každou cenu nutné, aby si ho čtenář povšiml a věnoval mu maximální možnou pozornost. Toho je dosaženo zmenšením šířky sazby a ve zbylém místě na okrajích je text výrazně orámován červenou (nebo černou v případě nastavení třídy dokument fastthesis na černobílou sazbu přepínačem “prn”) barvou.

Je-li z nějakého důvodu nutné změnit barvu postranních zvýrazňovacích pruhů, je možné tak učinit pomocí aparátu balíčku xcolor. Příkazem \definecolor se změní nastavení barvy pojmenované FASThesis@ImportantColor. Pokud tedy chce autor namísto předvolené červené zvýrazňovat důležitý text třeba zelenou barvou, učiní to třeba takto:

```
\definecolor{FASThesis@ImportantColor}{cmyk}{1.0, 0, 1.0, 0}
```

Netřeba dodávat, že výše uvedený příkaz se musí ve zdrojovém kódu dokumentu objevit **před** použitím prostředí important⁴.

3.7 Užitečné drobnosti

Třída fastthesis dává autorovi k dispozici ještě několik jednoduchých příkazů pro usnadnění práce (a také pro celkem žádoucí zvýšení podílu sémantické informace v textu). Výběr těchto pomocných příkazů je pochopitelně poplatný tomu, že primárními uživateli šablony jsou studenti FAV. Pokud zde nenajdete, co hledáte/potřebujete/očekáváte, vzhůru do studia jazyka \TeX u a vlastnoručního kódování \TeX ových maker (tedy příkazů).

Sazba významných/nově definovaných odborných termínů. K vyznačení důležitých nebo právě definovaných odborných termínů či zaváděných pojmů lze využít příkaz `\term{termín}`:

Jako `\term{stacionární bod}` funkce $f(x)$ označujeme takový bod a z~jejího definičního obodu, pro který platí, že $\frac{\mathrm{d}f}{\mathrm{d}x}(a) = 0$.

Následující odstavec ukazuje výsledek po zpracování \LaTeX em:

⁴Samozřejmě, že pokud se do toho pustíte, musíte také dočasně povolit používání znaku zavináč (“@”) jako běžného písmene (příkazy `\makeatletter` a `\makeatother`).

Jako **stacionární bod** funkce $f(x)$ označujeme takový bod a z jejího definičního obodu, pro který platí, že $\frac{df}{dx}(a) = 0$.

Sazba názvů souborů či cest v souborovém systému. Pro tento účel je připraven příkaz `\filename"název souboru"`, jehož parametrem je cesta v souborovém systému (tzv. **path**). Sazba zejména cest v operačním systému Windows je v \TeX u problematická, poněvadž \TeX používá jako uvozující znak příkazů zpětné lomítko ("`\`"), které ve Windows slouží jako oddělovač jednotlivých složek v cestě. Z toho důvodu není možné vysázet cestu běžným způsobem:

```
... \texttt{C:\Program Files\System32} ...
```

Na to reaguje \TeX chybovým hlášením, že nezná příkaz `\Program`. Řešení nabízí právě příkaz `\filename`, jehož použití je naznačeno v následujícím výpisu zdrojového kódu:

```
... \filename"C:\Program Files\System32" ...
```

Pozor: Tento příkaz (z implementačních důvodů) vyžaduje argument uzavřený mezi dvěma **stejnými** znaky, nikoliv mezi levou a pravou složenou závorkou, jak je to obvyklé. V roli „závorky“ lze použít jakýkoliv znak, který se nevyskytuje v argumentu, takže např. uvozovky jako v ukázce nebo třeba znak kůl ("`|`"), hvězdička, vykřičník atp.; třeba takto `\filename"C:\Program Files\System32"`.

Sazba příkazů či krátkých úseků zdrojového kódu. Obdobná situace nastává, je-li třeba vysázet některé úseky zdrojového kódu, které jsou natolik krátké, že nemá smysl je umisťovat do prostředí code (nebo obdobného na bázi `lstlisting`), ale zároveň obsahují znaky, které \TeX specifickým způsobem interpretuje. Příkladem může být třeba část kódu v MATLABu: `X = A \ B`. Tam se vyskytuje zpětné lomítko, které \TeX interpretuje jako uvozující znak příkazu.

K sazbě takových sekvencí znaků je určen příkaz `\command"zdrojový kód"`, v jehož parametru mohou být i speciální znaky, které by jinak \TeX interpretoval. Jeho použití je velmi přímočaré:

```
... \command"X = A \ B" ...
```

Opět platí, že roli „závorky“, ve které je argument uzavřen, může hrát jakýkoliv znak, který se pak ale v argumentu samozřejmě nesmí vyskytovat (a otevírací i uzavírací „závorka“ argumentu je stejná).

Sazba názvů či označení (softwarových) produktů. Obecně se považuje za vhodné opatřovat úseky textu dodatečnou sémantickou informací⁵ všude tam, kde je to možné a účelné. Takovou situaci může být třeba označování názvů různých výrobků,

⁵Tj. nějak určit, že právě tato sekvence znaků je třeba názvem nějakého produktu.

softwarových (např. **Inkscape** nebo **ReactOS**) nebo jiných produktů k tomu určeným příkazem `\product{název produktu}`, čímž jednak dojde k jejich zvýraznění v textu a jednak je možné s nimi dále pracovat (např. vytvořením jejich seznamu apod.):

Můj oblíbený textový editor `\product{Notepad++}` se hodí k~...

Sazba „tlačítek“. Při popisu např. grafického uživatelského rozhraní nějaké aplikace nebo třeba webové stránky se může hodit v textu graficky znázornit tlačítko, na které má uživatel kliknout apod. K tomu slouží příkaz `\button{text na tlačítku}`:

Po vyplnění formuláře klikněte na `\button{Odeslat}`.

Výsledkem výše uvedeného kódu je:

Po vyplnění formuláře klikněte na **Odeslat**.

Sazba kláves a klávesových zkratk. Při popisu ovládání software se často v dokumentech vyskytují sekvence kláves či jejich kombinací, které je třeba stisknout, aby proběhla požadovaná činnost. Třída `fastthesis` poskytuje k sazbě stisknutých kláves či jejich kombinací příkaz `\hotkey{označení kláves(y)}`:

Při psaní kvalifikační práce nezapomínejte na `\hotkey{Ctrl+S}`.

Výsledkem výše uvedeného kódu je:

Při psaní kvalifikační práce nezapomínejte na **Ctrl+S**.

Není to nic světoborného, ale důsledným používáním lze dosáhnout zejména tolik žádoucí (typo)grafické konzistence vytvářeného dokument. Navíc užíváním takovýchto jednoduchých příkazů (klidně si je vytvářejte i sami, je to dobrá věc) systematicky injektujeme další sémantickou informaci do textu (to je za všech okolností žádoucí) a pak třeba jen hledání, kde všude jsou uvedené nějaké klávesové zkratky, je výrazně snazší a rychlejší.

Lokální instalace fontu GT America



Třída dokumentu `fastthesis.cls` používá font **GT America** navržený nezávislou švýcarskou písmolijnou **Grilli Type**. Tento font je v *Manuálu jednotného vizuálního stylu Západočeské univerzity v Plzni* určen jako jeden ze dvou základních fontů, které vizuální identitu ZČU definují. Font ovšem není součástí instalace \TeX a ani pro něj neexistuje \TeX ový balíček. Je to proto, že **GT America** není volně dostupný – je třeba k němu mít licenci (kterou ZČU pochopitelně zakoupila, aby mohla font legálně používat).

Font se tedy musí doinstalovat, aby bylo možné třídu `fastthesis.cls` úspěšně využít k sazbě dokumentu. Není třeba se toho bát – \TeX se chová velmi „rozumně“, jen je nutné mu pár věcí vysvětlit. Postupujte podle níže uvedeného návodu:

1. V instalačním archivu `fastthesis.zip` je podsložka `install` a v ní pak podsložka `$TEXMFLOCAL`. Obsah této podsložky (jsou tam dvě složky `fonts` a `tex`) je třeba nakopírovat do kořenové složky lokální instalace \TeX u (to bývá např. v operačním systému **Windows** v případě distribuce MiKTeX složka `C:\Program Files\MiKTeX`).
2. Potom je třeba upravit tzv. **mapu fontů**. To se udělá příkazem `“initexmf --edit-config-file updmap”` v konzoli. Je-li \TeX nainstalován správně, spustí se po zadání tohoto příkazu předvolený systémový textový editor (v nejhorším případě **Notepad**) a v něm se otevře soubor `updmap.cfg` – tedy konfigurační soubor globální mapy \TeX ových fontů. Do tohoto konfiguračního souboru (na konec) musíme připsat řádek `“Map GTAmerica”`.
3. Po úpravě konfigurace mapy fontů je nutné celou mapu přegenerovat příkazem `“initexmf --mkmaps”`.
4. Pro jistotu ještě spustíme program `updmap`, který obnoví (pregeneruje) globální mapu fontů, příkazem `“updmap --verbose”`.

Přehled balíčků importovaných třídou `fastthesis`



Třída dokumentu `fastthesis` využívá (a tedy během své instanciaci úvodním příkazem dokumentu `\documentclass[...] {fastthesis}` zavádí) níže popsané \LaTeX ové balíčky:

1. `memoir` – třída dokumentu, od které je třída naší šablony `fastthesis` odvozena. Jedná se o velmi „mocnou“ třídu, která nahrazuje starší třídy \LaTeX u 2.09 ϵ `article/report/book`.
2. `inputenc` – zajišťuje správnou interpretaci znaků z rozšířených znakových sad (CP-1250, UTF-8 atp.) ve zdrojovém kódu dokumentu \LaTeX em.
3. `fontenc` – zajišťuje správné mapování znaků ve zdrojovém kódu dokumentu na *typy* (*písmové znaky*) použitého fontu.
4. `newtx` – balíček, umožňující \LaTeX u využívat fonty z projektu *TX Fonts*, tedy volně dostupné virtuální fontu odvozené od fontu *Adobe Times* a upravené pro \LaTeX .
5. `newtxmath` – viz předchozí, varianta pro sazbu matematických symbolů.
6. `microtype` – balíček tzv. *mikrotypografických* rozšíření, jejichž aplikaci v dokumentu umožnilo široké rozšíření pdf \LaTeX u. Dovoluje dokonale „vyladit“ sazbu použitým fontem např. adaptivním prostrkáním, rozpalem apod.
7. `babel` – rozsáhlý balíček podpory cizích jazyků (tedy např. češtiny, která je normálně \LaTeX u dost cizí).
8. `csquotes` – korektivní balíček zajišťující správnou sazbu českých uvozovek, tedy těch, jejichž uvozující znak leží na účarí a ukončující na horní dotažnici:
„...“

9. `biblatex` – balíček pro sazbu bibliografických informací na konci dokumentu a správu citací v textu.
10. `graphicx` – rozsáhlý a velmi „mocný“ balíček pro vkládání grafických objektů do dokumentu a manipulaci s nimi.
11. `xcolor` – balíček pro práci s barvami; umožňuje jednak barvy definovat v několika různých barevných prostorech a následně jimi obarvovat jednotlivé komponenty sazby (text, pozadí aj.).
12. `eso-pic` – balíček, který dovoluje vkládat grafické motivy na pozadí stránek.
13. `tcolorbox` – balíček, jehož pomocí lze obarvovat tzv. *boxy*, tedy sesazené (ohraničené nebo neohraničené) pravoúhlé oblasti naplněné textem.
14. `qrcode` – balíček, který umožňuje sázet QR kódy různých typů; v šabloně je využit pro (nepovinné) vysazení QR kódu s URL práce v informačním systému STAG na zadní stranu desek/přebalu.
15. `geometry` – balíček, který se mění resp. upravuje či přizpůsobuje geometrie stránky a tzv. *sazebné zrcadlo*, tedy např. vzdálenosti textu od okrajů, velikosti záhlaví a zápatí apod.
16. `tikz` – mohutný balík pro tvorbu a následnou sazbu vektorových ilustrací; pomocí speciálního jazyka na základě geometrických primitiv dovoluje popsat vzhled grafického motivu a ten posléze vložit do dokumentu.
17. `hyperref` – balíček, kterým se realizuje sazba aktivních hypertextových odkazů a také specifické typografické provedení jejich URL adres.
18. `titlesec` – balíček, s jehož pomocí lze upravit grafické provedení nadpisů jednotlivých částí dokumentu (kapitol, sekcí, podsekcí atp.).
19. `listings` – balíček s nástroji pro sazbu výpisů zdrojového kódu v mnoha různých programovacích jazycích, zvýrazňování syntaxe těchto jazyků apod.
20. `xparse` – balíček se sadou maker pro syntaktickou analýzu zdrojového kódu v jazyce \TeX ; třída `fastthesis` ho používá k realizaci některých maker, jejichž parametrem může být řetězec, obsahující znaky se speciálním významem, které by \TeX za normálních okolností nějak interpretoval (jako třeba zpětné lomítko).
21. `xstring` – balíček, pomocí kterého lze manipulovat s řetězci (obsahem proměnných, těly maker apod.). Lze např. (i) zjišťovat, zda řetězec začíná nebo končí nějakým znakem, (ii) zkoumat, zda řetězec obsahuje nějaký podřetězec atp.

- 22. `booktabs` – balíček, dovolující sázet komplikovanější a sofistikovanější tabulky, lépe a snáze než to lze standardními prostředky \LaTeX u.
- 23. `longtable` – viz předchozí, dovoluje tabulku rozdělit na více stránek.

Detailní informace ke všem balíčkům lze načerpat z jejich (obvykle velmi podrobných) dokumentací, které se všechny nachází v archivu zvaném **CTAN**, tedy **Comprehensive \TeX Archive Network**, na adrese URL <https://www.ctan.org>.

Autor dokumentu využívajícího šablonu `fastthesis` může tedy okamžitě a bez zavádění výše popsaných balíčků příkazem `\usepackage` do textu vkládat veškeré příkazy, které tyto balíčky poskytují.

Často kladené dotazy (FAQ)



Proč má výsledný dokument při použití této šablony dvě titulní stránky, jednu barevnou a jednu černobílou? — Protože ta první „titulní strana“ vlastně není titulní strana: Je to přední strana desek (nebo chcete-li přebalu). Současné technické možnosti maloprodukční vazby knih bez problémů dovolují je opatřovat plnobarevnými laminovanými deskami, které se mi zdají mnohem hezčí, než ty prastaré látkové či koženkové desky, do kterých se kvalifikační práce vkládaly již za Rakouska-Uherska. Proto šablona vytváří dokument tak, aby měl arch desek/přebalu, tj. přední a zadní stranu, plnobarevnou (příp. i s grafickým motivem) a autor si tak mohl nechat práci svázat do moderních plnobarevných desek.

Kdo o tuto variantu nestojí, prostě první a poslední stránku dokumentu (i) zahodí v případě, že je dokument již vytištěný, nebo (ii) vymaže v nějakém nástroji pro práci s dokumenty ve formátu PDF (třeba **Adobe Acrobat**, **PDF24**, nebo třeba jen terminálový **pdftk**).

Proč je v dokumentu tolik prázdných stránek? — Protože je primárně určen k tisku a svázání. Některé archy jsou ve vázané publikaci potištěné jen z jedné strany (např. titulka, patitul atp.) a konzervativní pravidla sazby také předepisují, že např. kapitola vždy začíná na liché (tedy při pohledu na otevřenou knihu) pravé stránce. Proto musejí být ve výsledném PDF prázdné stránky na těch pozicích, které jsou opačnými stranami archů potištěných jen z jedné strany.

Pokud hodláte dokument číst pouze v elektronické podobě a nebudete ho tisknout, jsou tyto prázdné stránky pochopitelně zbytečné. V takovém případě se jich lze snadno zbavit použitím přepínače „viewonly“ v inicializačním příkazu `\documentclass` – detaily najdete v tabulce 2.1 na str. 8.

Proč při překladu „vyskáče“ tolik varování, underfull a overfull boxů apod.? — Protože grafické provedení sázeného dokumentu definované touto šablonou je (na standardy \TeX u) velmi komplexní. Je zde celá řada tabulek, obrázků, výpisů zdrojového kódu atp., takže \TeX ovský algoritmus, který skládá boxy na stránku, se dostává

často na hranici svých možností a prostě se mu nedaří sesadit boxy lépe, tak aby skóre „ohavnosti“ bylo dostatečně nízké. Proto také trochu „nadává“... Nicméně to není jakosti výsledku příliš na závadu. Nenechte se tím rozhodit.

Šablona obsahuje chyby! Použil jsem příkaz `\xyz` a \TeX jen vyházel spoustu chybových hlášek! — Šablona byla před zveřejněním velmi intenzivně a velmi důkladně testována. Samozřejmě, že i tak mohla nějaká záludná chyba uniknout mé pozornosti (v takovém případě mne neváhejte kontaktovat), ale není to příliš pravděpodobné. Zkuste nejprve pořádně prohlédnout svůj kód, zda jste například nezapomněli u příslušného příkazu uvést některý z jeho povinných parametrů, zda jste nezapomněli někde uzavírací složenou závorku, zda jsou všechna prostředí začínající příkazem `\begin{...}` také ukončena odpovídajícím příkazem `\end{...}` apod.

Bibliografie

- [Bau22] BAU, Sheng. *Generation of a random graph*. TikZ.net, 2022-09-24. Dostupné také z: <https://tikz.net/generation-random-graph/>.
- [Ber+16] BERAN, Vladimír; ŠTORM, František; KOLENSKÝ, Vladimír; FASSATI, Tomáš. *Typografický manuál*. 8. vyd. Praha: Kafka design, 2016. ISBN 978-80-260-7606-3.
- [FF05] FIELL, Charlotte; FIELL, Peter. *Graphic Design Now*. Köln: Taschen, 2005. ISBN 978-3-8228-4778-7.
- [Pou12] POULIN, Richard. *Jazyk grafického designu*. Praha: Slovart, 2012. ISBN 978-80-7391-552-0.
- [Wil22] WILSON, Peter. *The Memoir Class for Configurable Typesetting*. The Herries Press, 2022. Dostupné také z: <https://texdoc.org/serve/memoir/0>. PDF, Version 3.7.19.

Seznam obrázků

3.1	Průběh funkce $f(x) = \sin(x) \cos(2x)$	22
3.2	Náhodný graf vysazený TikZem podle kódu Shenga Bau [Bau22] . . .	24

Seznam tabulek

2.1	Seznam všech použitelných přepínačů třídy dokumentu <code>fastthesis</code> . .	8
3.1	Přehled jednotek množství informace	25

Seznam výpisů

3.1	Ukázkový výpis kódu v jazyce ANSI C	27
3.2	Ukázkový výpis obsahu příkazového řádku Windows	29
3.3	Ukázkový výpis obsahu terminálu Linuxu	30

1101001 1100001
10101100001110010 1100001
101011010101 1100001



11010011101101001
01100001 1100001
1110001010111 1100001