

Gymnázium Brno, Vídeňská, příspěvková organizace



Ročníkový projekt  
pro Programování

# **Jak psát ročníkovou práci**

verze 2.5

16. dubna 2024

Ing. David Martinek



## **Abstrakt**

Tato práce se zabývá tím, jak psát ročníkové, maturitní, pololetní a podobné středoškolské projekty. V práci jsou nastíněny zásady tvorby strukturovaného dokumentu a shrnuta základní typografická pravidla. Dokument se také stručně zabývá problematikou používání literatury a citační etiky.

## **Klíčová slova**

typografie, sázení textu, ročníkový projekt

## **Citace**

Ing. David Martinek: Jak psát ročníkovou práci verze 2.5, ročníkový projekt, Brno, Gymnázium Brno, Vídeňská, příspěvková organizace, 2024.

# Jak psát ročníkovou práci verze 2.5

## Prohlášení

Prohlašuji, že jsem tuto práci vytvořil samostatně a poctivě jsem uvedl všechny zdroje, ze kterých jsem vycházel.

.....  
Ing. David Martinek  
16. dubna 2024

© Ing. David Martinek, 2024.

*Tato práce vznikla jako školní dílo na škole Gymnázium Brno, Vídeňská, příspěvková organizace. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je nezákon-  
né, s výjimkou zákonem definovaných případů. Autor souhlasí s archivací této práce a s jejím  
použitím pro studijní účely na výše zmíněné škole a jejích případných právních nástupcích.*

# Obsah

# Kapitola 1

## Zadání

Vytvoř vzorovou dokumentaci pro ročníkový projekt předmětu Programování. Specifikuj v ní minimální požadavky kladené na takovýto dokument. Popiš strukturu dokumentace, typografické požadavky a požadovaný rozsah.

Musí jít o strukturovaný dokument používající vhodným způsobem styly, číslované nadpisy, obsah atd. Pro vysázení dokumentu použij vhodný textový procesor typu  $\text{\LaTeX}$ , Libre Office a podobné.

# Kapitola 2

## Analýza a návrh řešení

### 2.1 Specifikace zadání

Zadání školního projektu je sám o sobě specifický literární žánr. Jeho účelem je specifikovat, co má řešitel projektu udělat a případně vytyčit směr, jakým má při řešení jít.

Vytyčí-li zadání směr řešení, neznamená to, že se jej řešitel musí za každou cenu držet. Zadavatel tím dává najevo, že očekává určitý způsob řešení. Řešitel se od něj může odchýlit, pokud to dobře zdůvodní. Ideální je, když řešitel tento svůj záměr předem konzultuje se zadavatelem.

Zadání musí být stručné a zároveň výstižné a jednoznačné. Příliš stručné zadání klade velké nároky na řešitele. Příliš nejednoznačné zadání může vést k výsledkům, které se rozcházejí s očekáváním zadavatele. A nakonec příliš rozsáhlé zadání zase obvykle nenechává příliš prostoru pro řešitelovu iniciativu, což je také špatně. Zadání také musí být realistické a odpovídat schopnostem, které jsou od řešitele očekávány.

Zadání by mělo být psáno rozkazovacím způsobem (prostuduj, navrhni, vytvoř, vyhodnoť výsledky). To platí i pro takový projekt, u něž má student možnost si zadání specifikovat sám. V tomto případě je nevhodné psát zadání v první osobě. Rovněž není vhodné slibovat (napíšu skvělý program...). Podmiňovat zadání (mohl bych napsat program, ...<sup>1</sup>) je již úplně mimo.

### 2.2 Analýza problémů

Pokud není zadání skutečně detailní (což by u školního projektu byla chyba zadavatele), je potřeba se nad ním zamyslet ještě před vlastním řešením. Především je potřeba prostudovat vhodnou literaturu na dané téma. Někdy bývá tento bod přímo specifikován v zadání. Podívat se do literatury je užitečné vždy, protože minimálně můžete zjistit, že někdo již dané téma zpracoval. V tom případě můžete na již hotový projekt nějakým způsobem navázat, vylepšit jej nebo jej udělat jinak a lépe. Veškerou prostudovanou literaturu je v dokumentaci potřeba poctivě citovat (viz dále).

To, že se nad zadáním zamýšlel, dá autor najevo tím, že v dokumentaci provede upřesnění původního zadání. Je potřeba identifikovat a zmínit problémy, které v zadání nejsou explicitně zmíněny, ale ze zadání vyplývají. Například když je v zadání napsáno, že máte vytvořit program, který zpracuje vstupní údaje ze souboru, kde budou uloženy ve vhodném formátu, vyplývá z toho několik problémů. Jak bude uživatel zadávat soubor se vstupy? Jak přesně má vypadat formát vstupního souboru, tedy co je správný vstup a co ne? Co se stane, když uživatel zadá vstupní soubor chybně a co, když bude vstupní soubor v chybném formátu? Tyto problémy je potřeba zmínit a stručně navrhnout jejich řešení ještě předtím, než se pustíte do vlastního programování.

---

<sup>1</sup>... zřejmě jen pokud se mi bude chtít.

## 2.3 Problémy při psaní dokumentace

Před započítím práce na dokumentaci je potřeba si uvědomit, jak má vypadat výsledný dokument a kdo a jak jej bude číst.

O tom, jak psát podobné typy školních prací napsal pěknou knihu například Umberto Eco (**Eco97**). Stojí za přečtení.

### Struktura dokumentu

Dokumentace musí být strukturovaná a členěná na sekce a kapitoly nikoli proto, že to chce učitel, ale proto, aby ji bylo možné číst rychle, přehledově a naopak, aby ji nebylo nutné číst sekvenčně. Dokumentace projektu má totiž charakter technické zprávy. Takový dokument se nečte jako krásná literatura, tedy od první až po poslední kapitolu, ale jako technický dokument, kdy čtenář chce vyhledávat klíčové části a chce je číst na přeskáčku.<sup>2</sup>

### Formát dokumentace

Vzhledem k výše napsanému, nelze dokumentaci tvořit jako nestrukturovaný textový dokument. I když dnes má dokumentace obvykle elektronickou podobu, požaduje se, aby ji i přesto šlo snadno vytisknout na papír formátu A4, aby měla vhodně nastavené tiskové okraje, čitelné písmo, číslované kapitoly a stránky a aby měla stránku se správně vygenerovaným obsahem. Dokumentace dále nesmí obsahovat pravopisné ani typografické chyby. Všechny tyto požadavky mají za cíl zvýšit čitelnost, srozumitelnost a použitelnost dokumentu.

S ohledem na zajištění těchto požadavků, na přenositelnost a snadnost vytištění, je dnes preferován formát PDF. Tento formát lze poměrně snadno generovat pomocí textových procesorů jako L<sup>A</sup>T<sub>E</sub>X (nebo jeho varianty X<sub>Y</sub>L<sup>A</sup>T<sub>E</sub>X, jako v tomto dokumentu), LibreOffice Writer, případně MS Office Word. Všechny tyto aplikace navíc podporují tvorbu skutečně strukturovaných dokumentů.<sup>3</sup>

### Rozsah dokumentace

Specifikovat požadovaný rozsah dokumentace v počtu vysázených stránek je naivní, zvláště požaduje-li se strukturovaný text. Například tento dokument bez samotného textu, pouze s nadpisy kapitol zabírá 7 stran.

Lepším řešením je počítat rozsah v tzv. normostranách. Jedna normostrana se počítá jako třicet řádků o šedesáti znacích včetně mezer. Přibližně to dělá 1800 znaků nebo 250 slov běžného textu. Obrázky se počítají jako jeden až dva odstavce textu, podle jejich informační hodnoty.

### Doporučené rozsahy

Uvedené rozsahy pro různé typy školních prací jsou pouze orientační. Když máte co říct, obvykle nemáte problém dosáhnout minimálního požadovaného rozsahu, ale naopak vlézt se do toho maximálního. Zde je nutné poznamenat, že příliš velký rozsah musí být dobře odůvodněný, jinak je stejným problémem, jako nedodržení minimálního rozsahu. Pokud máte tento problém, zvažte odsunutí méně podstatných detailů, grafů a obrázků do příloh.

---

<sup>2</sup>Takový manuál k práci nebo k televizi také nečtete jako román, ale hledáte v něm, co vás v danou chvíli zajímá.

<sup>3</sup>Ovšem pozor! I v textovém procesoru jde napsat nestrukturovaný dokument. Pouhé použití vhodného programu automaticky neznamená, že bude výsledek vysázen správně.



**Běžný ročníkový projekt** Zhruba 5 normostran. Jde o jednoduchý projekt bez velkých ambicí na skvělé hodnocení.

**Skvělý ročníkový projekt** Zhruba 10–15 normostran. Větší projekt s ambicí na lepší hodnocení a případné uplatnění při dalším studiu.

**Bakalářský projekt** Zhruba 20–30 normostran. Obvyklý požadavek na závěrečnou práci v nižším stupni vysoké školy. Na různých vysokých školách se mohou požadavky lišit.

**Diplomový projekt** Zhruba 40–50 normostran. Obvyklý požadavek na závěrečnou práci v magisterském stupni vysoké školy. Na různých vysokých školách se mohou požadavky lišit.

## 2.4 Typografické zásady

V českém strukturovaném textu je potřeba dodržovat typografické zásady pro český jazyk. Bez jejich dodržování bude dokument špatně čitelný a bude působit jako odbytý, což se jistě podepíše na celkovém dojmu.

Mezi hlavní typografické zásady patří:

- Dodržování pravidel českého pravopisu.<sup>4</sup> Překlepy kontrolujte korektorem pravopisu. Gramatická pravidla musíte znát sami. Chyby jako „by jsem“, „stane se to déle“ aj. za vás korektor pravopisu neodstraní spolehlivě.
- Odstavce jsou zarovnány do bloku. První odstavec za nadpisem nemá odsazen první řádek, ostatní jej odsazený mají (řešeno použitým stylem, nikoli nadbytečnými mezerami).
- Na koncích řádků nesmí zůstat jednoznačková slova, ani číslice. Řeší se to použitím nezlomitelných mezer (v  $\text{\LaTeX}$ u použitím příkazů `\~` a `\,.`).
- Dlouhá slova na koncích řádků se dělí na hranici slabik, přičemž platí předchozí pravidlo. Nelze dělit například takto: a-ontický, ú-voz.
- Věta nesmí začínat číslem a čísla nelze dělit na koncích řádků.
- Měrné jednotky se píší s nedělitelnou, ideálně zúženou mezerou (v  $\text{\LaTeX}$ u je to příkaz `\,`), když se čtou jako dvě slova (2 km – dva kilometry), kdežto bez mezery, když se čtou jednoslovně (2km – dvoukilometrový).
- Vyhýbat se cizím slovům na něž je naroubováno české skloňování. Například výrazy byte, byty, bytům lze smysluplněji používat v asimilované podobě bajt, bajty, bajtům, ...
- Kapitoly, obrázky, tabulky a vzorce jsou číslovány, aby se na ně dalo těmito čísly odkázat (viz sekce ??). Vzorce se číslují a odkazují čísla v kulatých závorkách (viz vzorec (??)). Do seznamu citované literatury se odkazuje číslem nebo značkou v hranatých závorkách (viz (**rybitex**)).
- Další zásady, jako psaní závorek, českých uvozovek, čárek mezi větami, čárek před spojkami a a nebo, výpustek, zkratk atd. jistě znáte z hodin českého jazyka.

## 2.5 Citace literatury

Pokud v dokumentaci ke svému projektu neodkazujete na žádnou literaturu, zřejmě to znamená, že buďto řešíte triviální projekt bez větších ambicí nebo si řešení takzvaně „cucáte z prstu“ nebo literaturu používáte, ale tajíte to.

<sup>4</sup>No jo, opravdu je potřeba je znát. Budou to po vás chtít i na vysokých školách a to i na těch technicky zaměřených.

Pokud žádnou literaturu nepoužíváte, zřejmě si zbytečně přiděláváte práci. U školních prací je velká pravděpodobnost, že dané téma již zpracoval někdo před vámi a pravděpodobně i lépe než vy. Kdybyste věnovali čas studiu literatury, zřejmě byste si ušetřili dost práce nebo mohli na nějaký hotový projekt navázat nebo jej zkusit řešit jinak. U středoškolských projektů se nepředpokládá, že přijdete s něčím výrazně novým a převratným. U těchto projektů je cílem prokázat jistou úroveň znalostí a schopností a v ideálním případě si vyzkoušet řešení zajímavého reálného problému.

Poslední možnost, kdy literaturu používáte, ale tajíte to, je nejhorší. Vystavujete se podezření, že vaše řešení nebo jeho část je ve skutečnosti opsaným plagiátem. Inspirovat se podobným řešením ještě neznamena jej opsat. Vždy máte možnost řešit problém trochu jinak, pokusit se původní řešení vylepšit nebo s ním polemizovat. Pokud takovou literaturu správně citujete, hrajete s otevřenými kartami a dáváte recenzentovi možnost se přesvědčit, že jste nic neopsali. V opačném případě recenzent většinou podvod snadno odhalí (protože také umí vyhledávat na internetu nebo v dřívějších projektech).

Specifickou kategorii tvoří citování internetových zdrojů, zvláště Wikipedie. U těchto zdrojů je potřeba vždy uvést, kdy jste naposledy webovou stránku navštívili (podívejte se na záznam (**Citace15**)). Dále je potřeba být vždy konkrétní. Nelze například citovat Wikipedii jako celek. To má informační hodnotu stejnou, jako kdybyste do dokumentu napsali „někde jsem to našel“. Je potřeba citovat konkrétní stránku.

U Wikipedie (ale i jiných internetových zdrojů) je na místě opatrnost, protože jde o poměrně dynamický zdroj, který může kdokoli upravovat, včetně různých vtipálků nebo přímo záškodníků. Může se snadno stát, že použijete informace ze stránky, která je právě ve stavu po zásahu vtipálka a před opravou seriózních editorů. Kvalitní články na Wikipedii mají svůj vlastní seznam literatury, ve které lze popisované informace ověřit. Obvykle se doporučuje citovat spíše tyto spolehlivější zdroje, pokud jde například o odkazy na recenzované vědecké články nebo knihy.

# Kapitola 3

## Popis řešení

### 3.1 Volba textového procesoru

Vzhledem k výše specifikovaným požadavkům je potřeba používat skutečně textový procesor, nestačí obyčejný textový editor. Z běžně používaných se nabízí textové procesory z kancelářských balíčků LibreOffice nebo MS Office nebo textové procesory T<sub>E</sub>X, L<sup>A</sup>T<sub>E</sub>X a jejich varianty.

Konkrétně tento dokument byl vysázen v L<sup>A</sup>T<sub>E</sub>Xu, přesněji v modernějším X<sub>Y</sub>L<sup>A</sup>T<sub>E</sub>Xu, který lépe zpracovává znaky v kódování UTF-8, umí používat systémové fonty (zde rodina fontů DejaVu Serif) a produkovat PDF, v němž je možné vyhledávat i v češtině.

Pokud psaní v L<sup>A</sup>T<sub>E</sub>Xu nezvládáte, lze základní dovednosti nalézt například v knize Jiřího Rybičky (**rybitex**). Také online program Overleaf (viz <https://www.overleaf.com/>), který je dnes velice populární pro vytváření L<sup>A</sup>T<sub>E</sub>Xových dokumentů, má obsáhlou a srozumitelnou dokumentaci plnou praktických ukázek. Pokud raději používáte kancelářské programy, samozřejmě je lze použít také, musíte ale důsledně používat styly a další nástroje pro tvorbu strukturovaných dokumentů.

### 3.2 Formát dokumentu

Tento dokument byl vysázen na papír velikosti A4, písmem velikosti 12 bodů a standardním řádkováním. Větší výchozí písmo a řádkování nedává moc smysl. Pokud nebylo větší písmo a řádkování explicitně požadováno, vystavujete se nebezpečí, že chcete uměle nafouknout počet stran.

Okraje stránek jsou nastaveny 2 cm po stranách, 3 cm nahoře pro záhlaví a 2 cm dole. Pokud by měl být dokument vytisknut a svázan, je možné nastavit vnitřní okraj o 1 cm větší.

### 3.3 Konkrétní struktura dokumentu

Struktura dokumentu odpovídá technické zprávě. Dokument začíná titulní stranou, následuje automaticky generovaný obsah, pak jednotlivé číslované kapitoly. Je zakončen seznamem citované literatury a případnými přílohami.

V X<sub>Y</sub>L<sup>A</sup>T<sub>E</sub>Xu jsem použil vlastní třídu gvidprojekt, která je založena na třídě memoir. Použitelná by byla i třída report. Ve Wordu nebo Writeru použijte také vhodnou šablonu nebo si upravte šablonu standardní. Nezapomeňte používat styly pro tvorbu nadpisů a odstavců, automatické číslování nadpisů, stránek, obrázků a vzorců a další nástroje pro sazbu.

Dokument se musí skládat minimálně ze tří částí. První část tvoří jasně formulované zadání. Druhá část se má zabývat analýzou problematiky a návrhem řešení, ovládání, případně vzhledu aplikace. Teprve třetí část se zabývá výsledným řešením.

Popis řešení nesmí obsahovat kopie částí hotového kódu. Zde je třeba popsat strukturu aplikace ve smyslu rozdělení na moduly a podprogramy. Dále musíte popsat vámi navržené datové struktury a podprogramy, které s nimi pracují. Tento popis musí být srozumitelný a hlavně popisovat význam datových struktur a podprogramů. Bylo by chybou popisovat program jako doslovný překlad z jazyka C do češtiny, stylem „jestliže je proměnná  $x$  větší než 5, zavolá se funkce  $xy$ , která na prvním řádku...“.

## 3.4 Typografie

Typografická pravidla pomáhají dodržovat různé nástroje textových procesorů, ale protože neznají význam textu, neumí automaticky hlídat všechno. Textové procesory například umí automaticky dělit slova na koncích řádků, ale neumí spolehlivě automaticky doplňovat nezlomitelné mezery mezi za jednopísmenná slova, aby nezůstávaly na koncích řádků. V MS Wordu se například nezlomitelná mezera vkládá pomocí zkratky CTRL+SHIFT+MEZERNÍK, v  $\text{\LaTeX}$ u se používá znak tilda  $\sim$ . Pro  $\text{\LaTeX}$  lze rovněž použít volně dostupný program `vlna`, který vkládá tyto mezery automaticky (ale je nutné je překontrolovat).

Dělení slov, číslování, sázení vzorců apod. dělají textové procesory samy, pokud používáte správné styly. Pro kontrolu překlepů slouží korektory pravopisu, gramatické chyby ale umí odhalit jen v omezené míře.

## 3.5 Číslování a odkazy

Nadpisy, stránky, obrázky a vzorce je potřeba číslovat automaticky. V  $\text{\LaTeX}$ u, Wordu i Writeru se to dělá použitím vhodných stylů. Pomocí čísel se dá na tyto elementy automaticky odkazovat. V  $\text{\LaTeX}$ u jsou k tomu určeny příkazy `\label{klic}` a `\ref{klic}`. Ve Wordu a Writeru je potřeba vložit automatické pole pro odkaz. Pokud byste odkazovali ručním zápisem čísla kapitoly nebo stránky, hrozí nebezpečí, že po dodatečných úpravách budete odkazovat na chybná místa. Automatické nástroje pro odkazování slouží k automatickému přepočítání odkazů.

Totéž platí pro generování obsahu. Nelze jej vytvářet ručně, musíte automaticky. V  $\text{\LaTeX}$ u jsem k tomu použil příkaz `\tableofcontents`, ve Wordu a Writeru najdete odpovídající příkaz v menu.

## 3.6 Obrázky a tabulky

Používejte především vektorové obrázky. Zvláště vkládáte-li schémata nebo grafy. Bitmapové obrázky jsou použitelné pouze, když potřebujete vkládat fotografie. Vektorové obrázky neztrácí kvalitu při zvětšování nebo jiných manipulacích s dokumentem.

Obrázky, stejně jako tabulky je nutné automaticky číslovat a lze se na něj v dokumentu odkazovat příkazem k tomu určeným. V  $\text{\LaTeX}$ u je to opět příkaz `\ref{klic}` (viz obrázek ??). U každého obrázku, či vložené tabulky by měl být slovní popis.

Číslované obrázky je vhodnější ukotvit mimo tok textu, nahoru nebo dolů na stránku. Když kotvíte obrázky k odstavcům, zaděláváte si na problémy, když se odstavec ocitne na konci stránky. V dokumentu pak vznikají nepěkné mezery. Na číslovaný obrázek se navíc lze odkázat odkudkoli.



Obrázek 3.1: Ukázka vloženého obrázku.

### 3.7 Vzorce

Větší vzorce se nepíší přímo do textu, ale jako samostatné odstavce. Vzorce se číslují čísly v kulatých závorkách na kraji stránky. Stejným číslem se pak na ně dá odkázat (viz vzorec (??)).

$$c = \sqrt{a^2 + b^2} \quad (3.1)$$

V  $\text{\LaTeX}$ u jsem pro sázení vzorce použil prostředí `equation`, Writer i Word mají grafické nástroje pro psaní vzorců myší.<sup>1</sup>

### 3.8 Citace literatury

V  $\text{\LaTeX}$ u jsem pro vygenerování seznamu literatury použil balík `biblatex`.<sup>2</sup> Databázi citovaných zdrojů zde tvoří jednoduchý textový soubor se strukturovaným popisem jednotlivých odkazovaných knih, článků a dalších. Tento formát je celosvětově tak rozšířený, že například u téměř všech vědeckých článků se na webu standardně vyskytuje i zápis ve formátu `BibTeX`, aby se na ně šlo snadno odkazovat.

V  $\text{\LaTeX}$ u jsem pro vygenerování seznamu literatury použil příkaz `\printbibliography` na konci dokumentu. Pro samotné odkázání do tohoto seznamu pak slouží příkaz `\cite{klíč}`, případně `\parencite{klíč}` – v závislosti na použitém stylu citací.

Ve Wordu a Writeru si musíte nejprve vytvořit databázi zdrojů pomocí zabudovaných nástrojů. Odkaz do ní se poté vkládá jako speciální pole, podobně jako odkaz na obrázek.

Formát položek v seznamu literatury specifikuje norma ČSN ISO 690. Platí však princip, že důležitější než formát je obsah, tedy nezatajování důležitých informací. Cílem je poskytnout čtenáři dostatek informací, aby citovaný zdroj mohl najít a přesvědčit se, že původní informace, na které jste odkazovali jsou správné. Raději citujte s chybami nežli vůbec.

Správný formát citací si můžete ověřit na webové stránce <http://www.citace.com>. Další informace o tom, jak správně citovat a co to je plagiátorství se dočtete například v publikaci (**Citace15**).

<sup>1</sup>Což se dá kvalifikovat jako týrání zvířat.

<sup>2</sup>Což je novější verze dříve používaného nástroje `BibTeX`. Narozdíl od něj ale umí citovat podle české normy ISO 690 a poradí si lépe s odkazy na web.

# Kapitola 4

## Závěr

Tento dokument má sloužit pro inspiraci, jak správně psát dokumentaci k ročníkovým projektům. Zároveň slouží jako návod či připomenutí, jak vytvářet některé části dokumentů. Minimálně maturanti by měli většinu věcí již znát, protože ovládání textového procesoru, základy typografie a český pravopis již jistě probírali v nižších ročnících.

Pamatujte, že dokumentaci je potřeba psát průběžně. Teoreticky byste měli mít první polovinu dokumentace, tedy část týkající se analýzy a návrhu řešení, ovládání a vzhledu, napsanou před samotným započítáním vlastního kódování projektu.

Z tohoto dokumentu můžete převzít hrubou strukturu kapitol. Nedělejte to ale mechanicky. Váš projekt je určitě specifický, jiný než můj. Mým úkolem bylo popsat, jak psát dokumentaci, vy máte zadáno určitě něco jiného.

Vy v závěru své dokumentace shrnete celou svou práci. Popište, co se povedlo a co ne. Nevymýšlejte si a nelžete. Netvrdte, že vaše řešení je skvělé a bez chyb, pokud opravdu takové není. Zároveň je zbytečné se předem omlouvat za všechny chyby, kterých jste se určitě dopustili, a kterých si zatím ani nejste vědomi.

Dále je užitečné se v závěru zamyslet, co by ve vašem projektu šlo zlepšit nebo jak by jej šlo rozšířit či na něj navázat. Touto informací dáte najevo, že o svém řešení přemýšlíte a zároveň usnadníte práci sobě, pokud budete na projekt navazovat příští rok, nebo dalším studentům, kteří si vaši práci přečtou a budou v ní chtít pokračovat či se jí inspirovat.