



Hodnocení vedoucího závěrečné práce

Vedoucí práce: Ing. Tomáš Pecka
Student: Jiří Folprecht
Název práce: Výukový nástroj pro LL a LR syntaktickou analýzu
Obor / specializace: Softwarové inženýrství 2021
Vytvořeno dne: 28. května 2024

Hodnotící kritéria

1. Splnění zadání

- [1] zadání splněno
- [2] zadání splněno s menšími výhradami
- ▶ [3] zadání splněno s většími výhradami
- [4] zadání nesplněno

Práci hodnotím jako průměrně náročnou. Student ale musel nastudovat téma LR analýzy, které se učí až v magisterském stupni studia.

Minimálně body o testování a rozšíření aplikace nejsou podle mě dostatečně splněny.

2. Písemná část práce

15 / 100 (F)

Textová část práce nespĺňuje nároky kladené na závěrečnou práci. Měla by být podrobnější a rozsáhlejší a také obsahovat více detailů, které jsou obvykle očekávány. Práce je velmi krátká (závěr je na straně 20). Text kapitoly 1 (Teoretická část) obsahuje mnoho nepřesností, překlepů, chybějících definic, definic, které jsou useknuté uprostřed věty, apod. Práce neobsahuje téměř žádnou diskusi cílů, neobsahuje žádné postupy z hlediska softwarového inženýrství (chybí analýza, sběr požadavků, případy užití, instalační příručka, ...). Práce obsahuje značné množství typografických chyb, např. sazba strany iv (a nejen sazba, i obsah), chybějící dolní indexy (např. def. 1.6), nevhodnou sazbu všech obrázků (bez použití float env., neexistují reference na obrázky), nevhodné typy písma (obzvláště kapitola 3), sirotci, apod. Styl citování je zvláštní, samotný odkaz na citovanou publikaci často tvoří samostatný odstavec. Několik sekcí obsahuje jen jednu větu. Po jazykové stránce občas věty nedávají smysl, např. chybí slova, nebo věty nejsou dokončené, chybí interpunkce, vyskytují se překlepy. Autor si občas plete např. terminální a neterminální symboly (hlavně v kapitole 3). Nedostatků faktických, jazykových i typografických je řada.

Kapitola 1, teoretická část, začíná tím, že si autor klade filosofickou otázku, co je vlastně symbol abecedy. Dále, definice 1.5 a 1.8 jsou nedokončeny. Sekce 1.2 definuje pojmy z

oblasti gramatik, často ale využívá dopředné reference. Je diskutabilní, zda def. 1.11 a 1.12 jsou vůbec definice. Definice syntaktické analýzy nejde udělat bez chybějící definice derivace větné formy v gramatice. Definice 1.16 a 1.17 je ta samá definice. Tvzení, že více rozkladů věty v gramatice je způsobeno zkracujícími pravidly je zavádějící, je to způsobeno nejednoznačností gramatiky (ano, mimo jiné i cykly v gramatice, které mohou být (mimo jiné) způsobeny zkracujícími pravidly). Sekce 1.4 a 1.5 s názvy LL a LR jsou uvedeny pár větami o tom, co znamenají zkratky LL a LR, což je jistě zajímavé, ale nikde není dána definice LL/LR gramatik/jazyků a konfliktů v gramatice. Algoritmus syntaktické analýzy není popsán ani pro jednu metodu. Úplně chybí souvislost se zásobníkovými automaty. V sekci 1.4 není jasné co je "tabulka LL(1)" a co znamená "dívat se o jeden symbol dopředu". Sekce 1.4 používá pro funkci first 3 notace (dokonce během 3 řádků pod sebou) a není jasné, zda jsou to různé funkce, nebo jsou všechny stejné. Algorithm 1 mixuje zvláště češtinu a angličtinu. Sekce 1.4.6 o $\$First_k\$$ vůbec nesouvisí s prací. Sekce 1.4.7 o konstrukci LL(1) tabulky popisuje jak se sestaví rozkladová tabulka a tím kapitola o LL parsování končí. Vůbec není vysvětleno jak se tabulka použije, k čemu slouží ani jak vypadá implementace syntaktického analyzátoru. V kapitole o LR autor úplně mlčí o existenci LALR parsovacích algoritmů, přestože je zmiňuje v sekci 3.9. Obrázek na straně 9 je bez popisku a kontextu nejasný. Zbytek kapitoly pokračuje v podobném duchu. Bohužel zde úplně chybí nějaké příklady, jak parsování probíhá, nebo jak se sestavuje množina LR položek. Z práce se bohužel o parsování moc dozvědět nedá.

Kapitola 2, existující řešení, popisuje dva existující nástroje, avšak mohla by obsahovat podrobnější analýzu a využití těchto nástrojů. Kapitola má 11 (slovy: jedenáct) řádků. Dobré podotknout, že samozřejmě existují i nástroje vizualizující LR analýzu, namátkou [1], ale v práci nejsou žádné takové zmíněny. Myslím, že ani rešerše nedosahuje kvality kladených na závěrečnou práci.

Kapitola 3 se věnuje implementaci. Autor popisuje datovou strukturu pro terminální a neterminální symbol. V jejich popisu je spousta chyb (zaměňuje terminály a neterminály, levé a pravé strany pravidla, apod.) Sekce 3.2 je ukončena uprostřed věty a není mi jasný její význam. Sekce 3.3 o použité knihovně immutable.js je docela rozsáhlá, v kontextu zbytku práce je až velmi překvapivě dlouhá. Sekce 3.4 a 3.5 o implementaci algoritmů jsou několikařádkové a není mi jasný jejich význam, jejich informační hodnota je nízká. Celé testování je shrnuto 3 (slovy: třemi) řádky v sekci 3.7, což mi nepřijde jako dostatečné splnění zadání. Jednotkové testy jsou určitě dobrý základ, ale co např. testování uživatelské? Testování, jak se implementace chová v různých prohlížečích? Testování výsledného DOM stránky?

V závěru autor píše "algoritmy určené pro složitější třídy gramatik jsou značně náročnější", ale neudává žádné hledisko, podle kterého tak tvrdí.

V seznamu bibliografie chybí seznam autorů u položky 5. Na tuto položku se také autor odkazuje také textem "dragonbook", který se sice používá v komunitě, ale nezasvěcenému nic neřekne.

Obsah příloh neseďí, v příloze žádný adresář se spustitelnou formou implementace není.

[1] <https://jsmachines.sourceforge.net/machines/lr1.html>

3. Nepísemná část, přílohy

60/100 (D)

Výpočet LL/LR automatů na první pohled funguje. Aplikace má několik nedostatků: Např. checkbox pro výběr parsovacího algoritmu ve výchozím stavu nemá zaškrtnuté nic a přesto aplikace počítá LL(1) rozkladovou tabulku. V módu LR(0) aplikace ignoruje simulaci

průběhu výpočtu levého rozkladu. V prohlížečích na bázi browseru chromium se tabulky renderují zvláště se scrollbar. Jednotlivé komponenty na obrazovce jsou velmi zvláště seskládány a velký prostor stránky je prostě prázdný i na poměrně standardním rozlišení displeje. Číslování pravidel v redukcích nesedí mezi jednotlivými tabulkami.

Aplikace měla být rozšířena a vylepšena oproti původní (viz seznam zdrojů v zadání). Sice přibyla možnost konstrukce LR automatů, nicméně z nějakého důvodu zmizela možnost si vypsat implementaci LL analýzy rekurzivním sestupem pomocí pseudokódu a možnost získat permalink pro daný stav aplikace. Design aplikace vypadá podobně jako původní (dokonce texty s nápovědou zůstaly stejné), ačkoliv autor očividně začal nový projekt. Z mého subjektivního pohledu je to krok směrem zpět.

Je škoda, že aplikace neukazuje množiny LR(k) položek. To mi přijde jako velký nedostatek. Autor zmiňuje, že to je možné budoucí rozšíření, myslím ale, že tohle by aplikace měla umět už teď.

V práci chybí skript pro sestavení produkční verze, musel jsem si ho dodat. Autor v README popisuje jen jak spustit aplikaci ve vývojářském módu. Aplikace vypisuje spoustu pro uživatele nesrozumitelných informací do konzole prohlížeče, pravděpodobně pozůstatky debugovacích výpisů z vývoje. Dovoluju si rozporovat autorovo tvrzení, že aplikace je uživatelsky přívětivá, neboť není podloženo žádným uživatelským testováním.

Kód jinak vypadá poměrně rozumně. Algoritmy jsou otestovány jednotkovými testy (které nejde ve výchozím stavu systému spustit, je potřeba zásah do nastavení projektu). Uvítal bych testy např. i prezentační vrstvy a CI/CD pipeline. Nic takového se ale v práci nenachází. Zvolená technologie, tedy client-side aplikace v prohlížeči, je zvolena vhodně.

4. Hodnocení výsledků, jejich využitelnost

55 / 100 (E)

Nasazení aplikace je možné, nicméně nedosahuje toho, co bylo očekáváno. Využití ve výuce bude problematické, neboť aplikace pozbyla některých svých vlastností a aktuální rozmístění prvků na obrazovce neumožňuje efektivně používat web při promítání. Chybí vizualizace LR(k) položek, což je důležitá součást výpočtu akčních a goto tabulek v LR analýze.

5. Aktivita studenta

- [1] výborná aktivita
- [2] velmi dobrá aktivita
- [3] průměrná aktivita
- [4] slabší, ale ještě dostatečná aktivita
- [5] nedostatečná aktivita

S autorem jsem moc v kontaktu nebyl. Konzultací bylo naprosté minimum a poslední proběhla na začátku letního semestru. Další kontakt byl 2 dny před odevzdáním, kdy mi autor poslal emailem odkaz na repozitář se zdrojovými kódy aplikace. Odevzdanou textovou část práce jsem poprvé obdržel až odevzdáním do systému projects, neměl jsem tedy možnost ji před odevzdáním vidět. Myslím, že důležitou schopností softwarového inženýra je i komunikace o zadání a cílech, která proběhla jen velmi okrajově.

Z historie změn ve verzovacím systému je ale vidět, že student nějakým způsobem pracoval na vývoji po celý letní semestr.

6. Samostatnost studenta

- [1] výborná samostatnost
- [2] velmi dobrá samostatnost
- [3] průměrná samostatnost
- [4] slabší, ale ještě dostatečná samostatnost
- [5] nedostatečná samostatnost

Jsem na rozpacích jak mám tento bod hodnotit. Student byl sice samostatný, neboť jsem o něm téměř celý letní semestr nevěděl a přesto výslednou práci odevzdal. Práci to ale tedy vůbec neprospělo, spíše by práci pomohlo, kdyby ji s někým konzultoval.

Celkové hodnocení

25 /100 (F)

Textová část práce podle mě nesplňuje požadované standardy kladené na závěrečné práce.

Implementační část vypadá funkčně, ačkoliv k ní mám nemalé výhrady.

Na základě těchto skutečností práci hodnotím stupněm F a nedoporučuji k obhajobě.

Instrukce

Splnění zadání

Posudte, zda předložená ZP dostatečně a v souladu se zadáním obsahově vymezuje cíle, správně je formuluje a v dostatečné kvalitě naplňuje. V komentáři uveďte body zadání, které nebyly splněny, posudte závažnost, dopady a případně i příčiny jednotlivých nedostatků. Pokud zadání svou náročností vybočuje ze standardů pro daný typ práce nebo student případně vypracoval ZP nad rámec zadání, popište, jak se to projevilo na požadované kvalitě splnění zadání a jakým způsobem toto ovlivnilo výsledné hodnocení.

Písemná část práce

Zhodnoťte přiměřenost rozsahu předložené ZP vzhledem k obsahu, tj. zda všechny části ZP jsou informačně bohaté a ZP neobsahuje zbytečné části. Dále posudte, zda předložená ZP je po věcné stránce v pořádku, případně vyskytují-li se v práci věcné chyby nebo nepřesnosti.

Zhodnoťte dále logickou strukturu ZP, návaznosti jednotlivých kapitol a pochopitelnost textu pro čtenáře. Posudte správnost používání formálních zápisů obsažených v práci. Posudte typografickou a jazykovou stránku ZP, viz Směrnice děkana č. 52/2021, článek 3.

Posudte, zda student využil a správně citoval relevantní zdroje. Ověřte, zda jsou všechny převzaté prvky řádně odlišeny od vlastních výsledků, zda nedošlo k porušení citační etiky a zda jsou bibliografické citace úplné a v souladu s citačními zvyklostmi a normami. Zhodnoťte, zda převzatý software a jiná autorská díla, byly v ZP použity v souladu s licenčními podmínkami.

Nepísemná část, přílohy

Dle charakteru práce se případně vyjádřete k nepísemné části ZP. Například: SW dílo – kvalita vytvořeného programu a vhodnost a přiměřenost technologií, které byly využité od vývoje až po nasazení. HW – funkční vzorek – použité technologie a nástroje, Výzkumná a experimentální práce – opakovatelnost experimentů.

Hodnocení výsledků, jejich využitelnost

Dle charakteru práce zhodnoťte možnosti nasazení výsledků práce v praxi nebo uveďte, zda výsledky ZP rozšiřují již publikované známé výsledky nebo přinášející zcela nové poznatky.

Aktivita studenta

V souvislosti s průběhem a výsledkem práce posudte, zda byl student během řešení aktivní, zda dodržoval dohodnuté termíny, jestli své řešení průběžně konzultoval a zda byl na konzultace dostatečně připraven.

Samostatnost studenta

V souvislosti s průběhem a výsledkem práce posudte schopnost studenta samostatně tvůrčí práce.

Celkové hodnocení

Shrňte stránky ZP, které nejvíce ovlivnily Vaše celkové hodnocení. Celkové hodnocení nemusí být aritmetickým průměrem či jinou hodnotou vypočtenou z hodnocení v předchozích jednotlivých kritériích. Obecně platí, že bezvadně splněné zadání je hodnoceno klasifikačním stupněm A.