



# Posudek oponenta závěrečné práce

Oponent práce:	Ing. Tomáš Pecka
Student:	Ondřej Štorc
Název práce:	Generovaný parser jazyka pro konzolového rozhraní knihovny algoritmů
Obor / specializace:	Webové a softwarové inženýrství, zaměření Softwarové inženýrství
Vytvořeno dne:	10. června 2023

## Hodnotící kritéria

### 1. Splnění zadání

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

Student splnil zadání a nad rámec zadání implementoval nové CLI a auto-completion.

### 2. Písemná část práce

72/100 (C)

V textové části autor popisuje základy parsování a poté použití nástroje ANTLR4 pro generování parserů, kterým autor bude nahrazovat ručně psaný parser, jehož výstupem bude (v projektu již existující) již existující abstraktní syntaktický strom. Dále v práci se autor věnuje problémům, na které narazil při výměně parseru za automaticky generovaný. Následuje kapitola věnující se testování implementace.

Práce je napsána v anglickém jazyce. Jazyková stránka je v pořádku. Po typografické stránce je práce celkem v pořádku. Na některých místech ale trošku ujíždí mezery mezi elementy (např. v matematické sazbě na str. 6), občas je na stránce málo obsahu společně s float prostředím, což vytváří dlouhé mezery mezi odstavci (např. na str. 26), občas jsem zahlédl i nevhodně použité pomlčky či spojovníky, či sirotky/vdovy.

Práce však definuje většinu pojmů jen neformálně a odkazuje čtenáře na jiné zdroje. Kapitola 2 (analýza) a hlavně sekce zabývající se teorií parsování jsou velmi stručné. V práci, která se zabývá hlavně teorií parsování a nástrojem ANTLR by bylo vhodné mít podrobnější analýzu. Na konci sekce 2.1 autor popisuje nejednoznačné gramatiky na příkladu dangling else problému, ale nezmiňuje jak tento problém ovlivňuje konstrukci příkazu if. V sekci 2.3.1 o konstrukci LL(1) parseru z LL(1) gramatiky např. autor definuje potřebné funkce First a Follow jen neformálně. Stejně tak rozkladovou tabulku. Z těchto

neformálních definic také není jasné jak se např. má parser chovat, je-li ve výhledu nečekaný symbol. Práci by také prospěla větší analýza parsovacího algoritmu ALL(\*), který je používán nástrojem ANTLR4. V sekci o ALL(\*) se také zmiňuje LL(\*) algoritmus, o kterém však v práci není žádná zmínka. Kapitola o jazyku AQL je také poměrně stručná a měla by dle mého názoru čtenáře s jazykem více seznámit.

Narazil jsem také na několik drobných chyb v převzatých definicích (např. záměna =>+ a =>\* v definici větné formy).

### 3. Nepísemná část, přílohy

99 /100 (A)

Autor implementoval automatické generování parseru jazyka aql. Gramatika i kód je přehledná. Autor se zorientoval v rozsáhlém a netriviálním C++ projektu a zvládl ho rozšířit o tuto novou funkcionalitu. Kód je vhodně a extenzivně otestován. Vítám autorovo rozhodnutí (i když velmi ovlivněné filosofií projektu ANTLR) o co největší rozdělení gramatiky a zpracování parse tree. Nad rámec zadání student také vyměnil původní knihovnu pro zpracování vstupu za novou, která vhodně zapadá do projektu. K tomu přidal i novou implementaci auto-completion.

Změny provedené autorem mají podrobnou a rozumnou historii ve verzovacím nástroji git, která usnadní orientaci v případě budoucích změn.

Snad jen drobnou výtkou může být, že v případě špatného vstupu se tento nepřidá do historie v CLI.

### 4. Hodnocení výsledků, jejich využitelnost

100 /100 (A)

Nový parser usnadní budoucí změny v gramatice jazyka aql. Nové CLI a auto-completion zase zvýší uživatelskou přívětivost. Všechny změny, které byly implementovány v rámci ZP, již byly přidány do projektu.

## Celkové hodnocení

90 /100 (A)

Písemná část práce je oproti implementační slabší, nicméně, vzhledem k tomu, že student implementoval i některé věci nad rámec zadání, hodnotím práci ještě 90 body, tedy stupněm A.

## Otázky k obhajobě

- Musel jste provést i jiné zpětně nekompatibilní změny, než změnu syntaxe pro cesty k souborům? Pokud ano, proč?

## **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.

### **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.