



Dokumentácia k projektu z predmetu IFJ a IAL

Implementácia interpreteru imperatívneho jazyka IFJ16

Tým 016, varianta b/4/II

5. decembra 2016

Riešitelia:

Sámel Šimon, xsamel02 ,
Patrik Sztefek, xsztef02 ,
Tomáš Szúcs, xszucs01 ,
Marek Šipoš, xsipos03 ,
Jakub Štol, xstolj00 ,

Obsah

<i>Dokumentácia k projektu z predmetu IFJ a IAL</i>	<i>2</i>
<i>Implementácia interpreteru imperatívneho jazyka IFJ16</i>	<i>2</i>
<i>Tým 016, varianta b/4/II</i>	<i>2</i>
<i>5. decembra 2016.....</i>	<i>2</i>

1. Úvod

V tejto dokumentácii je popísaný vývoj a implementácia interpreta imperatívneho jazyka IFJ16, ktorý je zjednodušenou podmnožinou jazyka Java SE 8. Celá dokumentácia je rozdelená do kapitol, v ktorých sú popísané jednotlivé časti interpreta. Od návrhu, implementácie až po samotnú prácu v tíme.

3. Vývoj interpretu

V tejto kapitole je popísane rozdelenie práce na interpretery, až po použité nástroje pri vývoji.

3.1 Rozdelenie práce

Prácu pred začatím vývoja interpretu rozdeľoval náš vedúci Patrik Sztefek. Rozdeľoval ju spôsobom, kto bol ako zručný v programovaní. Ale zároveň nám dal nám možnosť si vybrať, ktorú časť chceme riešiť. Aj keď sa môže zdať, že rozdelenie bodov neprináleží k odvedenej práci jednotlivých členov, každý sa snažil ako len mohol. Nižšie uvedené popisy slúžia pre predstavu, kto akú časť problematiky riešil.

Sámel Šimon – padlý v boji, stratený v labyrinte interpreta

Patrik Sztefek - Generátor, algoritmy, interpret, testovanie

Tomáš Szűcs – Lexikálny analyzátor, Chybové stavy, dokumentácia, občerstvenie, kazie morálky

Marek Šipoš – Syntaktický analyzátor, Sémantický analyzátor, testovanie, tímová panika

Jakub Štol - Lexikálny analyzátor, vstavané funkcie, výpomoc

3.2 Komunikácia a schôdze

Už pred začatím vývoja sme vedeli, že nebude čas na pravidelne schôdze z dôvodu odlišnosti rozvrhu každého člena, tak sme sa stretávali len vo vážnych prípadoch. Namiesto schôdze sme zvolili variantu Facebook-ovej skupiny a Facebook instant-messaging. Výhoda spočíva v tom, že vieme komunikovať z ktoréhokoľvek miesta. Zároveň podávať správu o pokroku na svojej práci.

3.3 Použité nástroje

Na vývoj interpretu sa použil verzovací systém **Git**. Ako úložný priestor pre náš projekt poslúžil **GitHub**. Z dôvodu dostupnosti na internete a zároveň má zabudované funkcie. Ktoré sprehľadňujú zmeny, a to v konkrétnom súbore na konkrétnom riadku. Aj z dôvodu poskytnutia voľnej licencie pre študentov, aj kvôli chuti vyskúšať moderné trendy. Ako ďalší nástroj sme použili **Google docs**. Najviac užitočný bol vo fázach vývoja, keď sme mali obrovské množstvo hlavičkových a zdrojových súborov. Zároveň slúžil ako poznámkový blok, či už pre brainstorming alebo popis jednotlivých častí interpretu.

3.4 Metodika vývoja softwaru

Ranné fázy vývoja by sa dali prirovnať k V-modelu, či už z pohľadu testovania alebo samotnej implementácie. Z dôvodu, že každá časť interpretu sa navrhla, implementovala následne testovala.

Príloha A: Konečný automat Lexikálneho analyzátoru



