|  |  |
| --- | --- |
| etf_logo_2010_cir | Univerzitet u Beogradu  Elektrotehnički fakultet  Programski prevodioci 1 (13S114PP1) |

Projekat

Kompajler za Mikrojavu

Predmetni profesor:  
prof. dr Dragan Bojić

Predmetni asistenti:  
as. ms Maja Vukasović  
as. ms Mihajlo Ogrizović

Student:  
Lazar Premović 2019/0091

Beograd, februar 2023.

Sadržaj

**No table of contents entries found.**

# Postavka projekta

Cilj projektnog zadatka je realizacija kompajlera za programski jezik Mikrojava.

Mikrojava je programski jezik nalik na Javu, ali znatno pojednostavljen.

Mikrojava između ostalog podržava i sledeće konstrukte:

* Globalne i lokalne promenljive, globalne funkcije, klase sa poljima i metodima
* Nadjačavanje metoda, supstituciju i polimorfizam
* Naredbe za kontrolu toka (if, if else, while, foreach)
* Naredbu za otpakivanje niza

Sam proces prevođenja je podeljen u četiri faze:

* Leksička analiza
* Sintaksna analiza
* Semanticka analiza
* Generisanje koda

Rezultujuci prevodilac je i sam podeljen na funkcionalne celine po ugledu na gorenavedenu podelu po fazama.

# Pokretanje projekta

Za prevođenje samog projekta je koriščen Ant build system (koji moze biti pokrenut komandom ’ant’ u korenom direktorijumu projekta), gde je celokupan proces podeljen u nekoliko koraka koji izvršavaju sledeće komande u datom redosledu:

1. Korak: delete
   1. rm .\src\rs\ac\bg\etf\pp1\Yylex.java
   2. rm .\src\rs\ac\bg\etf\pp1\MJParser.java
   3. rm .\src\rs\ac\bg\etf\pp1\sym.java
   4. rm .\src\rs\ac\bg\etf\pp1\\*.class
   5. rm .\src\rs\ac\bg\etf\pp1\util\\*.class
   6. rm -r .\src\rs\ac\bg\etf\pp1\ast\
   7. rm .\spec\mjparser\_astbuild.cup
2. Korak: parserGen
   1. cd src
   2. java -jar ..\lib\cup\_v10k.jar -destdir rs\ac\bg\etf\pp1 -ast rs.ac.bg.etf.pp1.ast -parser MJParser -buildtree ..\spec\mjparser.cup
   3. cd ..
3. Korak: lexerGen
   1. java -jar .\lib\JFlex.jar -d .\src\rs\ac\bg\etf\pp1 .\spec\mjlexer.flex
4. Korak: compile
   1. javac -classpath lib\\* -sourcepath src .\src\rs\ac\bg\etf\pp1\Compiler.java

Nakon toga preveden projekat se može pokrenuti skriptom compile.bat gde je prvi argument putanja do fajla sa Mikrojava programom a drugi argument putanja na koju će biti sačuvan izlazni objektni fajl.

Sadržaj skripte compile.bat:

java -cp .\src;.\config;.\lib\\* rs.ac.bg.etf.pp1.Compiler %1 %2

Preveden program se može pokrenuti, debagovati i disasemblirati pokretanjem skripti run.bat, debug.bat i disasm respektivno, sve skripte primaju putanju do objektnog fajla kao prvi argument.

Sadržaj skripte run.bat:

java -cp .\lib\mj-runtime-1.1.jar rs.etf.pp1.mj.runtime.Run %1

Sadržaj skripte debug.bat:

java -cp .\lib\mj-runtime-1.1.jar rs.etf.pp1.mj.runtime.Run -debug %1

Sadržaj skripte disasm.bat:

java -cp .\lib\mj-runtime-1.1.jar rs.etf.pp1.mj.runtime.disasm %1

# Opis test primera

Pored javnih test primera test301,test302 i test303, takođe su dostavljeni i sledeći test primeri:

* syntax-errors, koji testira sintaksne greške i oporavak od njih
* semantic-errors, koji testira semantičke greške
* list, koji testira generator koda na primeru ulančane liste
* unpack, koji testira naredbu otpakivanja niza

# Novouvedene klase

Ne računajući klase automatski generisane od strane leksera i parsera, uvedeno je osam novih klasa:

* Compiler, koja predstavlja glavni program i upravlja procesom prevođenja.
* EST, koja je izvedena is klase Tab i predstavlja nadogradnje na dostavljenu tabelu simbola.
* SemanticAnalyzer, koja je izvedena is klase VisitorAdaptor i implementira posetioce potrebne za semantičku analizu.
* SemanticAnalysisHelper, koja sadrži uslužne metode potrebne pri semantičkoj analizi.
* CodeGenerator, koja je izvedena is klase VisitorAdaptor i implementira posetioce potrebne za generisanje koda.
* CodeGenerationHelper, koja sadrži uslužne metode potrebne pri generisanju koda.
* LoggingUtils, koja sadrži uslužne metode koje olakšavaju logovanje podataka.
* SymbolUtils, koja sadrži uslužne metode koje olakšavaju prikaz simbola pri leksičkoj analizi.