**DOKUMENTACIJA**

Programski prevodioci - predmetni zadatak

**Osnovni podaci**

|  |  |  |
| --- | --- | --- |
| Broj indeksa | Ime i prezime | Tema |
| SW37/2019 | Stefan Milošević | Sintaksa i semantika programskog jezika java |

**Korišćeni alati**

|  |  |
| --- | --- |
| Naziv | Verzija |
| bison | 3.8.2 |
| JFlex | 1.7.0 |
| javac | 11.0.15 |

## 

## Evidencija implementiranog dela

Zadatak projektnog zadatka je bio implementacija sintaksne i semantičke analize mini jave. Implementirano rešenje se bazira na analizi jednog .java fajla sa glavnom klasom u njemu. Podržani su konstruktori, metode, varijable klase, pozivanje metoda unutar drugih metoda, korišćenje ključne reči this za korišćenje atributa klase

.

Glavni delovi:

* Leksička analiza
* Sintaksna analiza
  + Glavna klasa
  + Glavna metoda
  + Metode i varijable klase
  + Konstruktori
* Semantička analiza
  + Provera redefinisanja metode, varijable, konstruktora
  + Provera adekvatnog tipa
  + Provera naziva konstruktora
  + Provera vidljivosti varijabli(scope)
  + Provera poziva funkcija

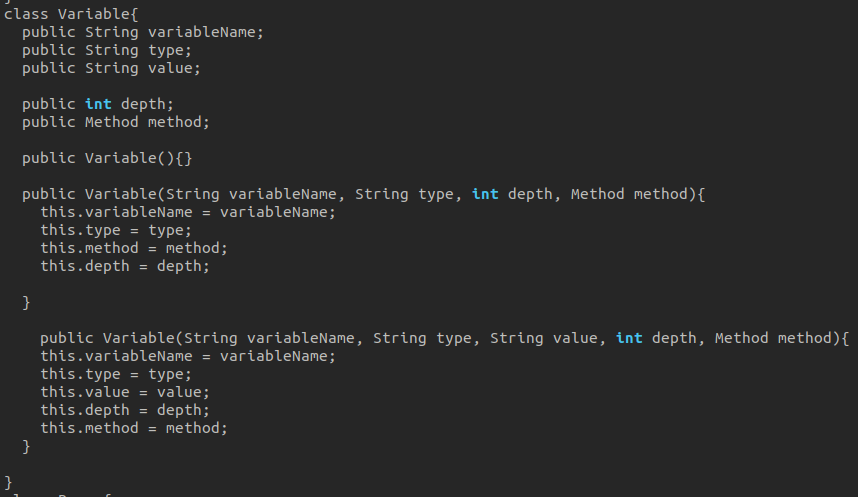
## Detalji implementacije

Leksička analiza se pokreće pomoću JFlex-a. Struktura .l datoteke je malo izmenjena u odnosu na standardni Flex, ali je suština ista.

Sintaksna i semantička analiza se pokreću pomoću Bison-a, specijalizovanog za Java programski jezik. U ovom slučaju je struktura .y fajla drugačija u odnosu na osnovni Bison, takođe, sav programski kod je napisan u Javi. U prvom delu datoteke se nalaze importi neophodnih java biblioteka. U drugom delu se nalazi programski kod klase parsera. Poljima i metodama ove klase može se pristupiti direktno iz dela gde su definisana gramatičkih pravila, koja se nalaze u sledećem segmentu .y datoteke. U ovom delu je instancirana tabela simbola, a deklarisane su metode i varijable koje će se koristiti za semantičku analizu. U main metodi se pokreće parser koji na kraju ispisuje uspešnost sintaksne i semantičke analize. U trećem segmentu .y datoteke nalazi se sama gramatika sa semantičkim proverama. U poslednjem segmentu nalazi se dodatni Java kod u kom se nalazi Lexer klasa (Slika 1), gde su definisane metode za vraćanje vrednosti i broja linije tokena. U ovom delu su, takođe, definisane potrebne klase: MainClass, Method, Variable, Constructor, Value, Param (Slika 2).



Slika Lexer klasa



Slika Variable klasa.

## Ideje za nastavak

Navedene stavke predstavljaju moguća poboljšanja:

* Mogućnost korišćenja klasnih metoda i varijabli iznad mesta u kome su deklarisani. Ovo bi zahtevalo pamćenje svih varijabli i metoda koje nisu deklarisane, kao i akcije koje se obavljaju. Nakon prvog prolaska parsera, još jednom bi se proverilo da li su označene metode i varijable deklarisane, kao i da li su akcije adekvatne.
* Poziv konstruktora unutar drugog konstruktora

## Literatura

Java bison - [Java Bison Interface (Bison 3.8.1) (gnu.org)](https://www.gnu.org/software/bison/manual/html_node/Java-Bison-Interface.html)

JFlex - [JFlex - JFlex User’s Manual](https://jflex.de/manual.html)

StackOverflow - [Stack Overflow - Where Developers Learn, Share, & Build Careers](https://stackoverflow.com/)