**DOKUMENTACIJA**

Programski prevodioci - predmetni zadatak

**Osnovni podaci**

|  |  |  |
| --- | --- | --- |
| Broj indeksa | Ime i prezime | Tema |
| SW31/2019 | Milan Ajder | Sintaksa i semantika java jezika |

**Korišćeni alati**

|  |  |
| --- | --- |
| Naziv | Verzija |
| Flex, Bison | Sa vežbi |

## Evidencija implementiranog dela

1. Klasa - sintaksna i semantička analiza
2. Konstruktor - sintaksna i semantička analiza
3. Atributi - sintaksna i semantička analiza
4. Metode - sintaksna i semantička analiza
5. Enkapsulacija - sintaksna analiza
6. Static - sintaksna analiza
7. Insatnciranje objekta – sintaksna i semantička analiza
8. If, else - sintaksna i semantička analiza
9. For - sintaksna i semantička analiza

## Detalji implementacije

1. Klasa – (kind – CLS)

Sintaksa – Klasa počinje sa enkapsulacijom, zatim ide specijalna reč class, naziv klase, otvorena vitičasta zagrada, sadržaj klase i zatvorena vitičasta zagrada. Sadržaj klase počinje sa definisanjem atributa, zatim ide definisanje konstruktora i na kraju definisanje metoda.

Semantika – Klasa mora da ima jedinstveno ime u fajlu. Ako je klasa privatna, svi atributi, konstruktori i metode se brišu iz tabele simbola.

1. Konstruktor (kind - CONSTR)

Sintaksa – Konstruktor počinje imenom klase, zatim ide otvorena zagrada, nula, jedan ili više parametara istog tipa, zatvorena zagrada, otvorena vitičasta zagrada, sadržaj konstruktora i na kraju zatvorena vitičasta zagrada. Sadržaj konstruktora može da bude prazan ili da ima jednu ili više linija. Linija je se sastoji iz specijalne reči this, zatim ide tačka, pa atribut klase, znak jednakosti, promeljiva koja se prosledi konstruktoru i na kraju tačka zarez.

Semantika – Konstruktor mora da bude unutar odgovarajuće klase. Parametri moraju da budu istog tipa. Tip parametra u konstruktoru ne sme biti void.

1. Atributi (kind – CLS\_ATR)

Sintaksa – Struktura: tip promenljive, lista promenljivih, tačka zarez.

Semantika – tip promenljive ne može da bude void. Promenljiva ne sme biti deklarisana više od jednom u odeljku.

1. Metode

Semantika – Struktura: enkapsulacija, sa ili bez static, tip povratne vrednosti funkcije, naziv funkcije, otvorena zagrada, tip parametara, ime parametra, zatvorena zagrada, otvorena vitičasta zagrada, telo funkcije, zatvorena vitičasta zagrada.

Sintaksa – funkcija mora da ima jedinstveno ime u klasi. Ako je funkcija privatna, parametri iz tela funkcije se brišu iz tabele simbola. Promenljive koje se koriste unutar funkcije moraju biti deklarisane unutar date funkcije ili da su atributi klase. Ako je funkcija void, onda ili nema return ili može da ima prazan return. Ako nije void onda mora da sadrži return sa promenljivom.

1. Enkapsulacija

U defs.h access\_specifier { PUBLIC=0, PRIVATE=1; }. Klase i funkcije mogu da imaju enkapsulaciju

1. Static

Funkcija može da ima static pre tipa povrtne vrednosti funkcije

1. Insatnciranje objekta

Sintaksa – Struktura: ime klase, ime promenljive, znak jednakosti, specijalna reč new, ime klase, otvorena zagrada, parametri prosleđeni konstruktoru odvojeni zarezom, zatvorena zagrada, tačka zarez.

Semantika: Ime klase i sa leve i sa desne strane jednakosti mora biti ista, naziv promenljive mora biti jedinstven, u datoj funkciji. Konstruktor može da primi samo parametre koji su deklarisani u datoj funkciji ili u klasi.

1. If/else

Ista kao implementacija sa časa

1. For

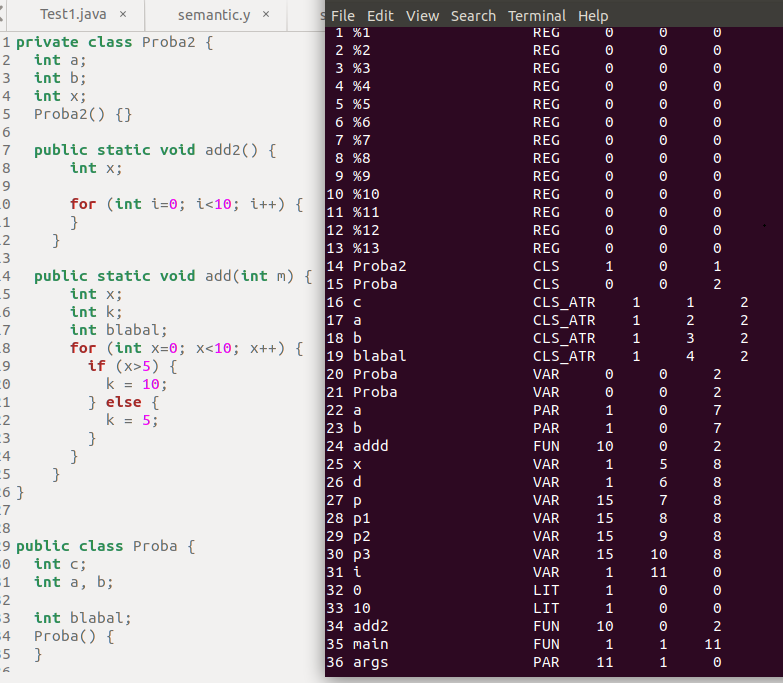
Sintaksa: ista kao i sa časa

Semantika: unutar petlje se može naći promenljiva koja je definisana unutar petlje, unutar funkcije ili klase.

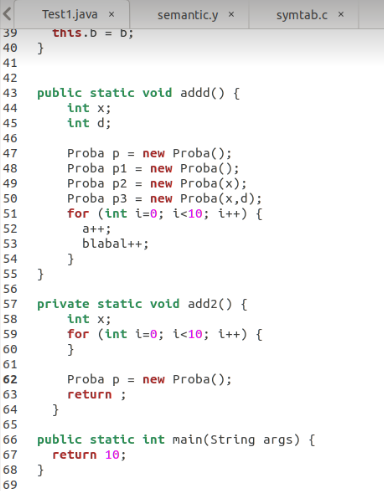
## Ideje za nastavak

Kako su sintaksa i semantika Jave dosta opširni, nisu implementirane sve funkcionalnosti. Dalji radi bi podrazumevao dodavanje semantičkih provera tamo gde nedostaju, dodavanje enkapsulacije za promenljive, nasleđivanja, interfejsa, enumeracija, switch/case, while petlje...

## Primer



Slika 1-tabela simbola za Test1.java



Slika 2 – kraj testa Test1.java

Primetimo da se iz tabele simbola brišu promenljive koje su definisane unutar privatnih funkcija. Isto važi i za klase.

## Literatura

Materijali sa predavanja: https://github.com/PPFTN/Osnove-proceduralnog-generisanja-pokreta-letnji-2021-22

Materijali sa vežbi: https://github.com/PPFTN/Programski-prevodioci-vezbe-letnji-2021-22-