Domaći zadatak iz Programskih Prevodioca 1

Veljko Rvović 2018/0132

Opis zadatka

Cilj projektnog zadatka je realizacija kompajlera za programski jezik Mikrojava. Kompajler  
omogućava prevođenje sintaksno i semantički ispravnih Mikrojava programa u Mikrojava bajtkod koji se izvršava na virtuelnoj mašini za Mikrojavu. Programski prevodilac za Mikrojavu ima četiri osnovne funkcionalnosti: leksičku analizu, sintaksnu analizu, semantičku analizu i generisanje koda.

Pokretanje projekta

Lexer, parser i ostali alat se kompajluje pokretanjem opcije ‘compile’ build.xml fajla. Ulazni fajlovi i testovi se pokreću menjanjem imena fajla unutar Compile.java fajla i pokretanjem sa opcijom run. Time se dobija izlazni objektni fajl čiji se sadržaj može pročitati pokretanjem opcije ‘disasm’ u build.xml fajlu, a može se i izvršavati sa opcijom ‘runObj’.

Kratak opis test primera

Sledeći test primer računa faktorijel brojeva od 1 do 5. Demonstrira rad petlji, nizova, i još nekih elemenata jezika.

program SuperTest

const int five = 5;

int i, j;

{

void main() int arr[]; {

arr = new int[5];

i = 0;

/\*

PROGRAM KOJI RACUNA i! za i=1,..,5

TESTIRANJE UGNJEZDENIH PETLJI I NIZOVA

TAKODJE VIDI FENSI MULTILINE KOMENTAR

\*/

do

{

arr[i] = 1;

j = i+1;

do

{

arr[i] = arr[i]\*j;

j--;

}while(j > 0);

print(arr[i]);

i++;

}while(i < five);

print(eol);

if( i == five) print('P');

}

}

Sledeći test primer demonstrira polimorfizam, konstrukciju objekata klasa kao i poziv konstruktora iz natklase.

program SuperTest

class A{

{

A(){ print('A'); }

}

}

class B extends A{

{

B() { super(); print('B'); }

}

}

class C extends B{

{

C() { super(); print('C'); }

}

}

{

void main() A a, a\_c; B b; C c;{

c = new C;

print(eol);

b = new B;

print(eol);

a = new A;

print(eol);

a\_c = new C;

}

}

Kratak opis novouvedenih klasa

Uvedene su klase SemanticAnalyer koji rade semantičku obradu programa i CodeGenerator koji na osnovu semantičkog stabla u jednom prolasku vrši generisanje koda.