
Elektrotehnički fakultet u Beogradu
Katedra za računarsku tehniku i informatiku

Predmet: Programski Prevodioci 1 (IR4PP1)
Nastavnik: doc dr Dragan Bojić
Asistent: dipl ing Miloš Gligorić
Ispitni rok: Oktobar 2008.
Datum: 28.09.2008.

Kandidat: _____

Broj Indeksa: _____ *E-mail:* _____

Ispit traje 3 sata, prvih sat vremena nije dozvoljeno napuštanje ispita. Upotreba literature nije dozvoljena.

<i>Zadatak 1</i>	_____ /8	<i>Zadatak 5</i>	_____ /10
<i>Zadatak 2</i>	_____ /8	<i>Zadatak 6</i>	_____ /5
<i>Zadatak 3</i>	_____ /6	<i>Zadatak 7</i>	_____ /6
<i>Zadatak 4</i>	_____ /9	<i>Zadatak 8</i>	_____ /8

Ukupno na ispitu: _____ /60 ***Ukupno na projektu:*** _____ /40

Rok u kome je odbranjen projekat: _____ (pr: jun 2008)

Ukupno: _____ /100

Ocena: _____ (_____)

Napomena: Ukoliko u zadatku nešto nije dovoljno precizno definisano, student treba da uvede razumnu pretpostavku, da je uokviri (da bi se lakše prepoznala prilikom ocenjivanja) i da nastavi da izgrađuje preostali deo svog odgovora na temeljima uvedene pretpostavke. Kod pitanja koja imaju ponuđene odgovore treba **samo zaokružiti** jedan odgovor. Na ostala pitanja odgovarati **čitko, kratko i precizno**.

1. [8] a) Navesti i ukratko prokomentarisati metode koje se pojavljuju u interfejsu tabele simbola.
b) Čemu služi klasa Scope i koje informacije ona sadrži?

2. [8] Vrste konflikata u LR(0) parsiranju. Objasniti kako se pregledom karakterističnog automata mogu u opštem slučaju uočiti konflikti.

3. [6] Pronađi suvišne neterminale u sledećoj gramatici sa startnim simbolom S. Prikazati detaljan postupak.

1. $\langle S \rangle \rightarrow a \langle S \rangle$	9. $\langle Q \rangle \rightarrow c \langle Q \rangle$
2. $\langle S \rangle \rightarrow a \langle B \rangle$	10. $\langle D \rangle \rightarrow a \langle E \rangle$
3. $\langle S \rangle \rightarrow c \langle E \rangle$	11. $\langle D \rangle \rightarrow c \langle Q \rangle$
4. $\langle B \rangle \rightarrow a \langle E \rangle$	12. $\langle D \rangle \rightarrow c \langle D \rangle$
5. $\langle B \rangle \rightarrow c \langle Q \rangle$	13. $\langle D \rangle \rightarrow \text{eps}$
6. $\langle B \rangle \rightarrow \text{eps}$	14. $\langle E \rangle \rightarrow a \langle E \rangle$
7. $\langle Q \rangle \rightarrow a \langle Q \rangle$	15. $\langle E \rangle \rightarrow c \langle E \rangle$
8. $\langle Q \rangle \rightarrow c \langle B \rangle$	

4. [9] Na narednoj slici, prikazane su potisna i kontrolna tabela LR(0) parsera. Na osnovu datih podataka potrebno je rekonstruisati karakteristični LR(0) automat. Objasniti.

	$\langle E \rangle$	id	(+)	-	
∇	$\langle E \rangle_{x1}$	id_x					SHIFT
$\langle E \rangle_{x1}$				$+_3$		$- _0$	SHIFT
$- _0$							ACCEPT
id_x			($_2$				SHIFT/REDUCE(1)
$+_3$		id_3					SHIFT
id_3							REDUCE(3)
($_2$	$\langle E \rangle_{x2}$	id_x					SHIFT
$\langle E \rangle_{x2}$				$+_3$	$)_2$		SHIFT
$)_2$							REDUCE(2)

5. [10] Data je L-atributivna gramatika ($\langle S \rangle_q$ – sintetizovano q ; $\langle A \rangle_{x,y,z}$ – nasledjeno x , sintetizovano x,y):

1. $\langle S \rangle_q \rightarrow a_{x1} \langle A \rangle_{x,y,z} \langle S \rangle_{q1} \{b_w\}$
 $w \leftarrow y^*z + q1, x \leftarrow x1, q \leftarrow q1$
2. $\langle S \rangle_{r1} \rightarrow b a_y \{c_w\} \langle A \rangle_{u,r,x} \{b_v\}$
 $w \leftarrow y+3, r1 \leftarrow r, v \leftarrow r^*x-w, u \leftarrow w+2$
3. $\langle A \rangle_{x,y,z1} \rightarrow a_q \{c_v\} \langle A \rangle_{x1,z,u} \langle A \rangle_{u,t,y1}$
 $v \leftarrow x-q, x1 \leftarrow x, y \leftarrow y1, z1 \leftarrow z, u \leftarrow 3$
4. $\langle A \rangle_{t,s,s1} \rightarrow b a_q \{b_{t1}\} a_{s2}$
 $t1 \leftarrow t, (s,s1) \leftarrow s2$

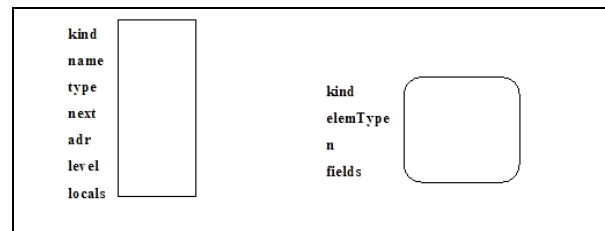
- a) [3] Odrediti FOLLOW skupove za svaki neterminal. Prikazati postupak.
- b) [3] Odrediti SELECT skupove za svaku smenu. Prikazati postupak.
- c) [4] Projektovati sintaksni procesor na bazi rekurzivnog spusta. Objasniti.

6. [5] Za prikazani MirkoJava program, nacrtati izgled tabele simbola u tačkama (**1**) i (**2**). Za umetanje umesto jednostruko ulančane liste koristiti heš tabelu (vredost heš funkcije određuje se na osnovu prvog slova imena: isto slovo imaće istu vrednost heš funkcije. Koristiti tehniku odvojenog ulančavanja: elementi sa istim vrednostima heš funkcije ulančavaju se u jednostruko ulančanu listu). Universe opseg nije potrebno crtati. Izgled Object i Struct čvora dat je kao potsetnik.

```

class ABC
char[] m;
int max;
char npp;
{
int mul (int x)
{
(** 1 **)
x++;
print(x, 5);
npp = 'C';
return x;
}
}
(** 2 **)

```



7. [6] Formirati sintakšno stablo za iskaz $g = a * x / (y+3) - a * b + 1$. Zatim generisati troadresni medjukod.

8. [8] U jednoj jednostavnoj implementaciji MJ interpretera memorijske oblasti opisane su unutar apstraktne klase Instruction, prikazane sledećim Java kodom (isečak):

```
public abstract class Instruction {
    /* oblast memorije koja sadrzi kod metoda */
    public static byte[] code;
    /* indeks instrukcije koja se izvrsava */
    public static int pc;

    /* staticki/globalni podaci glavnog programa */
    public static int[] staticData;

    /* dinamicki alocirani objekti i nizovi */
    public static int[] heap;
    /* pokazivac na pocetak slobodnog prostora */
    public static int free;

    /* procesorski stek */
    public static int[] procStack;

    /* stek izraza */
    public static int[] exprStack;
    /* indeks poslednje zauzete lokacije exprStack steka */
    public static int esp;

    public abstract void execute();
}
```

Za svaku MJ bytecode instrukciju postoji odgovarajuća podklasa koja implementira promenu stanja MJ VM (u skladu sa instrukcijom), kada se pozove metoda execute.

Potrebno je izvršiti interpretiranje (odgovarajuću promenu stanja) za sledeće instrukcije: putfield s, new s. Kostur odgovarajućih klasa dat je u nastavku:

```
public class PUTFIELD extends Instruction {
    // operand instrukcije putfield s
```

```
private char s;  
  
    // --- IMPLEMENTACIJA ---  
}  
  
public class NEW extends Instruction {  
    // operand instrukcije new s  
    private char s;  
  
    // --- IMPLEMENTACIJA ---  
}
```

Pretpostaviti da su sve memorijske oblasti inicijalizovane kao i data polja (operandi) prikazanih klasa.

Napomena: Detaljno prokomentarisati kod.