

---

---

Elektrotehnički fakultet u Beogradu  
Katedra za računarsku tehniku i informatiku

*Predmet:* Programski Prevodioci 1 (IR4PP1)  
Programski Prevodioci (RI4PP)

*Nastavnik:* Prof. dr Dragan Bojić

*Ispitni rok:* Februar 2008.

*Datum:* 01.03.2008.

*Napomene:*

- *Ispit traje 3 sata. Prvih 30 minuta nije dozvoljeno napuštanje sale. Prvih sat i 30 minuta nije dozvoljeno iznositi ispitne zadatke. Poslednjih 10 minuta ispita nije dozvoljeno napuštanje sale.*
- *Na ispitnoj vežbanci obavezno naznačiti broj osvojenih poena na domaćem zadatku u sledećem formatu:*

*Rok u kome je odbranjen domaći \_\_\_\_\_/(npr: jun 2007)*

*Ukupno poena \_\_\_\_\_/45*

- *Na ispitnoj vežbanci potrebno je naznačiti pitanja i zadatke koji su radjeni (zaokruživanjem odgovarajćeg broja).*
- *Nepoštovanje nekog od pravila povlači negativne poene.*

---

*Pitanja*

1. [8] Objasniti podelu aktivnosti kod dvoprolaznih prevodilaca. Nabrojati prednosti i mane dvoprolaznih prevodilaca u odnosu na jednoprolazne.
2. [8] Definicija pravilne LR(1) konfiguracije. U kom odnosu sa LALR(1) parserom je pomenuta konfiguracija?

## Zadaci

3. [11]

a. [2] Da li su dati automati ekvivalentni. Prikazati detaljan postupak.

	0	1	2	
Ax	Ax	Ax	Bx	0
Bx	Bx	Ax	Bx	1

	0	1	2	
Az	Cz	Az	Bz	0
Bz	Bz	Az	Az	1
Cz	Cz	Az	Bz	0

b. [5] Konvertovati regularni izraz:  $(b|ac)^*jx^*$  u nedeterministički automat (stanja automata obeležavati sa Sb gde b uzima vrednosti od 0). Koristiti Tompsonov algoritam.

c. [4] Nedeterministički automat dobijen pod b prebaciti u deterministički.

4. [4] Projektovati potisni automat koji prepoznaje sledeće skupove sekvenci:  $\{0^m 1^n\}$   $n > m \geq 0$ . Objasniti.

5. [8]

a. [3] Dodeliti attribute sledećoj gramatici tako da neterminalni simbol  $\langle \text{num} \rangle$  ima sintetizovan atribut jednak vrednosti oktalnog broja koji ovaj simbol generiše (iz oktalnog u decimalni sistem). Pretpostaviti da ulazni simbol OCT ima svoj atribut koji je jednak vrednosti odgovarajuće cifre (0, 1, 2, 3, 4, 5, 6, 7).

$\langle \text{num} \rangle \rightarrow \langle \text{num} \rangle \text{OCT}$

$\langle \text{num} \rangle \rightarrow \text{OCT}$

b. [5] Implementirati TD parser na bazi rekurzivnog spusta za sledeću AT gramatiku:

1.  $\langle \text{integer} \rangle_{\text{value}} \rightarrow \text{DIGIT}_{\text{val}} \langle \text{more\_digits} \rangle_{v, b}$   
 $\text{value} \leftarrow \text{val} * 10^b + v$

2.  $\langle \text{more\_digits} \rangle_{v, b} \rightarrow \text{DIGIT}_{\text{val}} \langle \text{more\_digits} \rangle_{v1, b1}$   
 $v \leftarrow \text{val} * 10^{b1} + v1 \quad b \leftarrow b1 + 1$

3.  $\langle \text{more\_digits} \rangle_{v, b} \rightarrow \varepsilon$   
 $v \leftarrow 0 \quad b \leftarrow 0$

6. [4] Za dati troadresni kod napisati implementaciju putem indirektnih trojki.

```
t1 := *p
t2 := i - 1;
b[t2] := t1;
t3 := j + 1;
j := t3;
t4 := a[j]
*p := t4;
```

7. [12] Dat je sledeći deo programa na jeziku sličnom Pascal-u:

```
procedure main;
  var ma, mb: integer;
  procedure procA(p1: integer, p2: integer);
    var a: integer;
    procedure procB;
      begin
        procA(63, 23);
      end;
    procedure procC;
      begin
        procB;
      end;
    begin
      procC;
    end;
  begin
    mb := 4;
    procA(ma, 13);
  end.
```

- a. [3] Koliko je naredbi potrebno za pristup nelokalnoj promenljivoj putem displeja. Dati primer.
- b. [5] Ako je statičko okruženje za nelokalne promenljive realizovano preko displeja, nacrtati izgled steka nakon formiranja aktivacionog zapisa za prvi poziv procA(63, 23). Dispelji se nalaze unutar aktivacionog zapisa. Za proceduru main kreira se aktivacioni zapis. Procedura main je na leksičkom nivou 1.
- c. [4] Napisati 80x86 asemblerski potprogram ekvivalentan datom programu za slučaj b) ako se koriste naredbe enter i leave.

8. [5] Za dati Mikro Java bytecode treba napisati Mikro Java kod (.mj). Imena klasa trebaju da počinju slovom c, imena konstanti nizom karaktera con, imena globalnih promenljivih slovom g, imena lokalnih promenljivih i formalnih parametara slovom l, imena polja slovom f, imena metoda slovom m. Smatrati da se u programu koriste samo složeni tipovi i tip int. Za sve promenljive za koje nije poznat tip usvojiti int.

```

0. enter 2, 3
1. getstatic 1
2. const 53
3. add
4. load 0
5. add
6. putstatic 1
7. load 1
8. exit
9. return
10. enter 0,3
11. new 8
12. putstatic 2
13. getstatic 2
14. getfield 0
15. getstatic 2
16. getfield 1
17. add
18. store 2
19. exit
20. return

```

load b	... → ..., val
putstatic s	..., val → ...
getstatic s	... → ..., val
putfield s	..., adr, val → ...
const w	... → ..., val
add	..., val1, val2 → ..., val1+val2
new s	... → ..., adr
aload	..., adr, index → ..., val