Elektrotehnički fakultet u Beogradu

Predmet:	Programski prevodioci 1
Nastavnik:	dr Dragan Bojić, red. prof.
Asistenti:	doc. dr Maja Vukasović mast.inž. Mihajlo Ogrizović mast.inž. Luka Hrvačević
Ispitni rok:	Avgust 2026.
Datum:	01.09.2026.

Potpis dežurnog

Kandidat: _____

Broj Indeksa: _____ *Smer:* _____ *Sala:* _____

Ispit traje 150 minuta.
Nije dozvoljeno korišćenje literature.
Prvih sat vremena nije dozvoljeno napuštati ispit.

Zadatak 1	_____/10	*Zadatak 4*	_____/10
Zadatak 2	_____/10	*Zadatak 5*	_____/10
Zadatak 3	_____/10	*Zadatak 6*	_____/10

Ispit: _____/60 **Ukupno:** _____/100

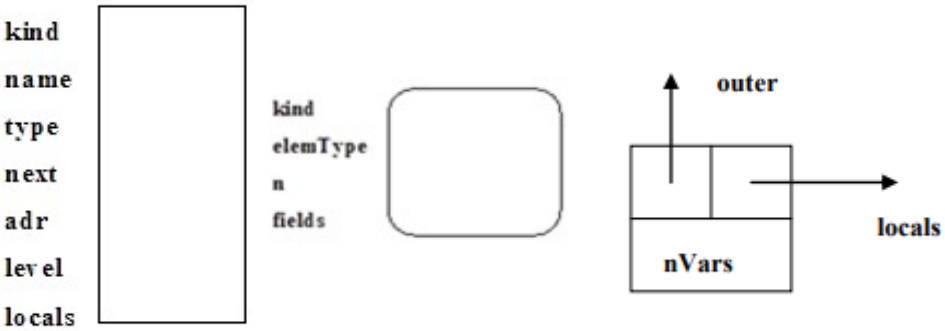
Projekat: _____/40 **Ocena:** _____

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. Na pitanja odgovarati **čitko i precizno**. Srećno!

Podsetnik za neke instrukcije Mikrojava bajtkoda

getstatic	s	... → ..., global[s]
putstatic	s	..., val → ...
new	s	... → ..., adr
getfield	s	..., adr → ..., adr.fields[s]
putfield	s	..., adr, val → ...
const	w	... → ..., w
load	b	... → ..., local[b]
store	b	..., val → ...
new	s	... ..., adr
newarray	b	..., n ..., adr
aload		..., adr, index ..., val
astore		..., adr, index, val ...
baload		..., adr, index ..., val
bastore		..., adr, index, val ...
enter b1, b2		
dup		..., val ..., val, val
dup2		..., v1, v2 ..., v1, v2, v1, v2
dup_x1	..,val2, val1	...,val1, val2, val1
dup_x2	val1, val2, val3	...,val3, val1, val2, val3 ...

Podsetnik strukture čvorova tabele simbola.



1) **(10 poena)**
Konstruisati potisni automat koji prihvata sve sekvence nad alfabetom $\{a,b,c\}$ za koje važi $\#a + \#b = \#c$, gde $\#x$ označava broj pojavljivanja simbola x u ulaznoj sekvenci, a redosled simbola je proizvoljan.

Rešenje:

2) **(10 poena)**
Dat je sledeći programski fragment:

```
int a = 1;
int b = 4;
int c = 0;

while (a < b) {
    if (c < a) {
        c = c + b;
        a++;
    } else {
        b = b - 1;
    }

    if (c > 10)
        break;
}

c = a + b + c;
```

a) Za dati programski fragment napisati troadresni međukod.
b) Dobijeni međukod predstaviti u vidu grafa toka kontrole na nivou bazičnih blokova u SSA formi.

Rešenje:

3) **(10 poena)**

Data je sledeća gramatika:

<S> → <S> a
<S> → b <A>
<S> → Ɛ
<A> → c <A> b
<A> → c <A> d <A>
<A> → Ɛ

a) [5] Nalaženjem SELECT skupova proveriti da li je data gramatika tipa LL(1). Ako nije transformisati je u LL(1).
b) [5] Korišćenjem metode rekurzivnog spusta realizovati parser nad gramatikom dobijenom pod a).

Rešenje:

4) **(10 poena)**

Za gramatiku iz zadatka 3 konstruisati:

a. potisnu i kontrolnu tabelu LR(0) parsera
b. kontrolnu tabelu SLR(1) parsera.

Rešenje:

5) **(10 poena)**

Na osnovu zadatog determinističkog konačnog automata A konstruisati minimalni deterministički automat B. Automat B treba da prihvata sve sekvence tipa $(xy|yx)^+$, gde su x sve sekvence koje automat A prihvata, a y sve sekvence koje automat A ne prihvata.

	1	0	
→A		B	0
B	C		0
C		A	1

Automat A

Rešenje:

6) **(10 poena)**

U mikrojavu se uvodi nova jezička konstrukcija `range` deklaracija unutar petlje, slično Python-u:

```
Statement = "foreach" “(" ident "in" "range" "(" Expr₁ "," Expr₂ ")" ")" Statement
```

Tokom semantičke analize u `CUP` akcijama (koristeći standardni `Tab` modul mikrojave) potrebno je proveriti sledeće kontekstne uslove:

1. $Expr_1$ i $Expr_2$ moraju biti tipa `int`.
2. Identifikator `ident` mora biti prethodno deklarisana lokalna ili globalna promenljiva (ili formalni parametar) isključivo tipa `int`.
3. `ident` ne sme biti konstanta (`Obj.Con`) niti ime tipa (`Obj.Type`).

a) Napisati pseudo-kod unutar CUP akcije koji pronalazi `ident` u tabeli simbola (`Tab.find`) i vrši sve tri semantičke provere, ispisujući adekvatnu poruku o grešci u slučaju neuspeha. Pretpostaviti da ident ima atribut tipa stringa, a Expr neterminali po jedan atribut tipa referenca na Struct objekat. Ne koristi se AST nego "on the fly" evaluacija.

b) Napisati kostur bajtkoda koji kompajler treba da emituje za navedenu foreach konstrukciju. Indeks petlje ident treba da uzima vrednosti od $Expr_1$ do $Expr_2$-1 sa korakom 1. Ispitivanje uslova je pre izvršavanja tela petljle, tako da se petlja neće nijednom izvršiti ako je $Expr_1 \geq Expr_2$.
Napomena: samo čist bajtkod assembler, ne traži se korišćenje klase Code. Jednostavnosti radi, pretpostaviti da je ident lokalna promenljiva sa indeksom 0 i ne razmatrati ostale varijante.

Rešenje:

Dodatni prostor za rad: