
Elektrotehnički fakultet u Beogradu

Predmet: Programski prevodioci 1
Nastavnik: dr Dragan Bojić, red. prof.
Asistenti: Maja Vukasović, mast .inž.
Kristijan Žiža, mast. inž.
Ispitni rok: Februar, 2021.
Datum: 12.2.2021.

Potpis dežurnog

Februarski ispitni rok iz Programskih prevodilaca 1

Kandidat: _____

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

*Ispit traje 120 minuta.
Nije dozvoljeno korišćenje literature.
Prvih sat vremena nije dozvoljeno napuštati ispit.*

1.	2.	3.	4.	5.	Σ

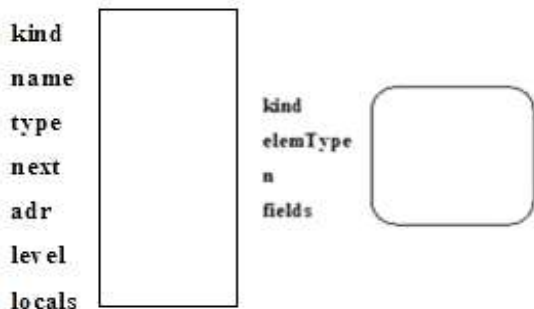
Napomena: Ukoliko u zadatku nešto nije dovoljno precizno definisano, student treba da uvede razumnu pretpostavku, da je uokviri (da bi se lakše prepoznala prilikom ocenjivanja) i da nastavi da izgrađuje preostali deo svog odgovora na temeljima uvedene pretpostavke. Na pitanja odgovarati **čitko i precizno**. 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) (12 poena)

Data je sekvenca od nula ili više promenljivih razdvojenih zarezima. Svaka promenljiva predstavlja niz od jednog ili više znakova, počinje obavezno malim slovom, nakon čega može doći prozvoljan broj znakova iz skupa koji čine mala slova, cifre i specijalni karakter “_”.

- a) Napisati regularni izraz koji opisuje navedenu sekvencu.
- b) Na osnovu regularnog izraza pod a) formirati minimalni DKA korišćenjem metoda pozicije.

Rešenje:

2) (12 poena)

Za datu gramatiku

1. $\langle S \rangle \rightarrow \langle A \rangle \langle S \rangle$

5. $\langle B \rangle \rightarrow a \langle D \rangle$

2. $\langle S \rangle \rightarrow \varepsilon$

6. $\langle C \rangle \rightarrow a \langle D \rangle$

3. $\langle A \rangle \rightarrow \langle B \rangle b \langle B \rangle c$

7. $\langle D \rangle \rightarrow \varepsilon$

4. $\langle A \rangle \rightarrow \langle C \rangle c \langle B \rangle$

- a) Nacrtati stanje ∇ karakterističnog LR(0) automata i vrstu ∇ kontrolne tabele SLR(1) parsera. Ima li konflikata?
- b) Nacrtati stanje ∇ karakterističnog LALR(1) automata i vrstu ∇ kontrolne tabele LALR(1) parsera. Ima li konflikata?
- Napomena: ne crtati ostatak karakterističnog automata, niti ostatak kontrolne tabele.

Rešenje:

3) (12 poena)

1. $\langle S \rangle \rightarrow a \langle A \rangle$
2. $\langle S \rangle \rightarrow \langle S \rangle b$
3. $\langle A \rangle \rightarrow c \langle A \rangle$
4. $\langle A \rangle \rightarrow d$

Na ulaz LR(0) parsera koji je opisan datom gramatikom dovedi se ulazna sekvenca **accbcadb**.
Pretpostavka je da se za oporavak od grešaka koristi jednostavan panic mode algoritam:

- a) sa sigurnim simbolom **c**.
- b) sa sigurnim simbolom **d**.

Za tačku a) i b) **zasebno** prikazati rad parsera i jasno naznačiti da li je parser uspešno završio parsiranje.

Rešenje:

4) (12 poena)

Kompajler za mikrojavu na izlazu daje sledeći ispis tabele simbola i listing bajtkoda. Kako izgleda izvorni mikrojava program koji odgovara ovim izlazima?

```
=====SADRŽAJ TABELE SIMBOLA=====
(Level -1)
Type int: int, 0, 0
Type char: char, 0, 0
Con eol: char, 10, 0
Con null: Class, 0, 0
Meth chr: char, 0, 1 [Var i: int, 0, 1 ]
Meth ord: int, 0, 1 [Var ch: char, 0, 1 ]
Meth len: int, 0, 1 [Var arr: Arr of notype, 0, 1 ]
Prog PROBA: notype, 0, 0
[Type SUPERKLASA: Class, 0, 0 [Fld s: int, 1, 1 ][Meth foo: int, 0, 1 [Var this: Class, 0, 2 ]]]
[Type POTKLASA: Class, 6, 0 [Fld s: int, 1, 1 ][Fld p: int, 2, 1 ][Meth foo: int, 9, 1 [Var this: Class, 0, 2 ]]]
[Var sk: Class, 12, 0 ]
[Meth main: notype, 19, 0 ]
```

Mikrojava bajkod:

0: enter 1 1	27: putstatic 0	74: const 111	119: putfield 0
3: const_0	30: const 111	79: putstatic 7	122: putstatic 12
4: jmp 3 (=7)	35: putstatic 1	82: const 111	125: getstatic 12
7: exit	38: const 111	87: putstatic 8	128: const_1
8: return	43: putstatic 2	90: const_m1	129: putfield 1
9: enter 1 1	46: const_m1	91: putstatic 9	132: getstatic 12
12: const_1	47: putstatic 3	94: const 9	135: dup
13: neg	50: const 0	99: putstatic 10	136: getfield 0
14: jmp 3 (=17)	55: putstatic 4	102: const -2	139: invokevirtual foo
17: exit	58: const -2	107: putstatic 11	156: const_0
18: return	63: putstatic 5	110: new 12	157: print
19: enter 0 0	66: const 102	113: dup	158: exit
22: const 102	71: putstatic 6	114: const 6	159: return

Rešenje:

5) (12 poena)

Za dati programski fragment napisati odgovarajući međukod u SSA formi i nacrtati graf toka kontrole na nivou osnovnih blokova.

Rešenje:

```
x = 0;
y = 1;
z = 2;
do{
    x = x - 1;

    if(z == 0) break;
    else y = y + z;

    z = y + 1;
} while (z > 1 && x == 0);
x = z - y;
```

