

---

Elektrotehnički fakultet u Beogradu  
Katedra za računarsku tehniku i informatiku

*Predmet:* Testiranje softvera (SI3TS)  
*Nastavnik:* doc. dr Dragan Bojić  
*Asistent:* dipl. ing. Dražen Drašković  
*Ispitni rok:* Februar 2015.  
*Datum:* 13.02.2015.

*Kandidat\*:* \_\_\_\_\_

*Broj indeksa\*:* \_\_\_\_\_

*Ispit traje 3 sata, a u toku prvog sata nije dozvoljeno napuštanje ispita.  
Upotreba literature nije dozvoljena.*

<i>Zadatak 1</i>	_____ /5	<i>Zadatak 5</i>	_____ /10
<i>Zadatak 2</i>	_____ /5	<i>Zadatak 6</i>	_____ /10
<i>Zadatak 3</i>	_____ /8	<i>Zadatak 7</i>	_____ /10
<i>Zadatak 4</i>	_____ /12		

*Ukupno na ispitu:* \_\_\_\_\_ /60      *Ukupno na domaćem\*:* \_\_\_\_\_ /40

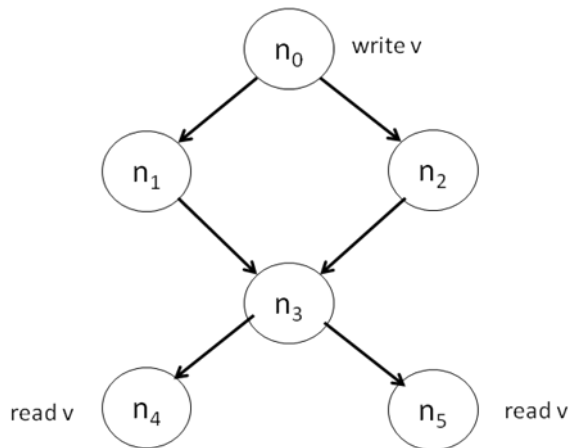
*Rok u kome je odbranjen domaći\*:* \_\_\_\_\_ (primer: januar 2015)

**Ukupno:** \_\_\_\_\_ /100

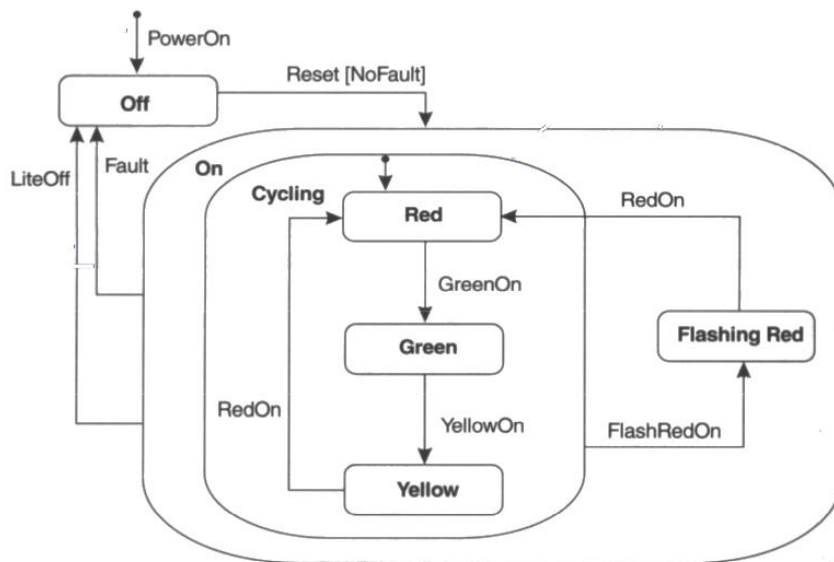
**Ocena:** \_\_\_\_\_ ( \_\_\_\_\_ )

**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**. \* popunjava student.

1. [5] Za dati programski graf odrediti putanje koje zadovoljavaju kriterijume:
  - a) All-defs
  - b) All-uses
  - c) All-du-paths



2. [5] Za zadati hijerarhijski dijagram stanja, nacrtati običan (nehijerarhijski).



3. [8] Data je java funkcija sa sledećom specifikacijom ponašanja: ako je  $x == \text{null}$  baciti *NullPointerException*, inače vratiti indeks poslednjeg elementa iz  $x$ , koji je jednak  $y$ . Ako takav element ne postoji, vratiti  $-1$ . Funkcija sadrži defekat.
- (a) Ako je moguće, kompletno specificirati test primer koji ne izvršava defektan kod.
  - (b) Ako je moguće, kompletno specificirati test primer koji izvršava defektan kod, ali ne detektuje grešku.
  - (c) Ako je moguće, kompletno specificirati test primer koji detektuje grešku.
  - (d) Objasniti gde je greška I kako je otkloniti.

```
public static int findLast(int[] x, int y) {  
    for (int i = x.length - 1; i > 0 ; i++) {  
        if ( x[i] == y ) {  
            return i;  
        }  
    }  
    return -1;  
}
```

4. [12] Dat je sledeći fragment koda sa skupom test primera.

```
if x<10 or y>50
    then S1
else S2
if w=50 or z>10
    then S3
else S4
```

Test primeri:

	x	y	w	z
<b>T1</b>	0	0	0	0
<b>T2</b>	11	75	50	12
<b>T3</b>	5	15	10	5
<b>T4</b>	15	75	50	3

Za svaki od sledećih kriterijuma testiranja nacrtati posebnu tabelu gde se po vrstama navode konkretni elementi programa koji treba da se pokriju razmatranim kriterijumom (na primer, za pokrivanje iskaza biće vrste S1, S2, S3, S4), i pored toga navode se oznake onih test primera koji pokrivaju posmatrani element programa (npr. u vrsti S1 biće T1, T2, T3, T4). Sračunati procenat pokrivenosti koda testovima za svaki od zadatih kriterijuma.

- Pokrivanje iskaza
  - Pokrivanje odluka
  - Pokrivanje elementarnih uslova
- Minimalno pokrivanje višestrukih uslova



5. [10] Potrebno je testirati veb aplikaciju “Kalkulator skijanja”, sa glavnim ekranom prikazanim na slici.

Implementacija programskog koda aplikacije nije poznata, pa je potrebno testirati aplikaciju na osnovu specifikacije zadatka:

Kada se klikne na sliku otvara se link skijalistasrbije.rs u novom prozoru. U padajućoj listi postoje sledeće opcije: Najam skija, Najam snowboarda, Najam opreme sa ski školom. Kada se unese ispravan datum rođenja, automatski treba da se popuni polje Kategorija: ako osoba ima manje od 18 godina, onda se popuni polje „dete“, ako ima između 18 i 25 onda se popuni polje „student“, ako ima između 26 i 64 onda se popuni polje „odrasla osoba“ i ako ima iznad 65 treba popuniti polje „senior“. Datum rođenja treba da bude u formatu dd/mm/gggg, a kontakt telefon u međunarodnom formatu + (dve-tri cifre) (razmak) (dve cifre) / (šest-sedam cifara).

Kada se klikne na dugme POTVRDI, treba otvoriti novi prozor u kome se ispisuje ime i prezime osobe koja je uneta u formu i tekst „Ukupno za uplatu“, koji se izračunava po sledećoj formuli:

$(CENA\_HOTELA + IZNAJMLJIVANJE\_OPREME) * POPUST$

gde je Iznajmljivanje opreme: najam skija = 3000din.; najam snowboarda = 4000 din.; najam opreme sa ski školom = 8000din. Popust se računa prema kategoriji osobe: za odrasle osobe nema popusta, za seniore 10%, za studente 20% i za decu 50% popust.

a) Metodom podele na klase ekvivalencije, kombinovano sa metodom graničnih slučajeva, testirati datu veb aplikaciju, pri čemu treba testirati i ulazne podatke i izlazne podatke (uspešan ispis ukupnog računa za uplatu ili ispisivanje neke poruke sa greškama, u vidu prozora upozorenja).

b) Metodom kombinatornog testiranja *All-pairs*, napisati skup test primera koji će testirati padajuću listu (iznajmljivanje opreme) i radio dugmad (kategorija korisnika).



6. [10] A) Data je Javina funkcija za brojanje karaktera u nekom tekstualnom fajlu.

a1) Odrediti sve LCSAJ za dati programski kod. Svaku sekvencu predstaviti kao uređenu trojku:

[POČETAK\_SEKVENCE, KRAJ\_SEKVENCE, LINIJA\_NA\_KOJU\_SKAČEMO]

a2) Odrediti minimalan skup testova koji pokrivaju sve LCSAJ iz tačke a). Za svaki test primer napisati šta je ulazni podatak i šta je očekivani izlazni podatak.

```
1. public class BrojacReci {
2.     public static void main(String[] args) {
3.         try {
4.             File f = new File("recnik.txt");
5.             Scanner sc;
6.             sc = new Scanner(f);
7.             sc.useDelimiter("[^a-zA-Z']+");
8.             Map<String, Integer> wordCount;
9.             wordCount = new TreeMap<>();
10.            while(sc.hasNext()) {
11.                String word = sc.next();
12.                if(!wordCount.containsKey(word))
13.                    wordCount.put(word, 1);
14.                else
15.                    wordCount.put(word, wordCount.get(word) + 1);
16.            }
17.            for(String word : wordCount.keySet())
18.                System.out.println(word + " " + wordCount.get(word));
19.            System.out.println(wordCount.size());
20.        }
21.        catch(FileNotFoundException e) {
22.            System.out.println("Problem prilikom citanja iz fajla.");
23.        }
24.    }
25. }
```

B) Neka je data matrica celih brojeva:

```
double zbir = 0;
int[][] matrica;
for(int i=0; i<m; i++)
    for(int j=0; j<n; j++)
        zbir+=matrica[i][j];
```

b1) Objasniti projmove živih, mrtvih i ekvivalentnih mutanata, i napisati test primere koji će prikazati njihovo postojanje, ukoliko koristimo mutacioni operator koji += mutira u -=, \*=, /=.

b2) Šta se dešava ukoliko koristimo operator OPPO, koji izraz x++ mutira u ++x i x-- (gde je x neka promenljiva)?





7. [10] Neka je data aplikacija *Trouglovi* (sa glavnim ekranom prikazanim na slici ispod). Koristeći FSM (*Finite State Machine*) testirati događaje koji se dobijaju klikom na određeni element prikazanog grafičkog korisničkog interfejsa. Koordinate temena biraju se iz padajućih listi koje imaju vrednosti uređenih parova koordinata  $(x, y)$  iz skupa svih celobrojnih vrednosti koordinata, počev od tačke  $(-5, -5)$  do tačke  $(5, 5)$ . Odabir temena je preduslov da može da se potvrdi dugme “Crtaj”, u suprotnom je to dugme onemogućeno (*disabled*). Kada se uspešno unesu temena, aplikacija preračunava dužine stranica, aktivira se dugme “Crtaj” i kada se potvrdi to dugme, iscrtava se trougao u koordinatnom sistemu, ukoliko on zadovoljava uslov postojanja nekog trougla (jednakostraničnog, jednakokrakog ili raznostranog).

Obim i površina su checkbox-ovi koji se odnose na mere trougla i mogu se čekirati oba, jedan ili nijedan checkbox. Kada se potvrdi dugme “Izračunaj”, u tekstualnim poljima ispod sistema se pojavljuju vrednosti obima i površine. Ukoliko nisu uneta sva temena trougla, a pritisne se dugme “Izračunaj” pojavljuje se greška (sa desne gornje slike). Trougao takođe ne može da se iscrtava, ukoliko ne ispunjava uslove postojanja trougla, i tada se pojavljuje greška (sa desne gornje slike), pritiskom na dugme “Crtaj” ili “Izračunaj”. Greške se javljaju kao *pop-up* prozori i imaju vremensko trajanje od 5 sekundi. Dugme “Poništi” vraća aplikaciju u prvobitno stanje (prazna polja za koordinate temena i nijedan checkbox nije odabran). Dugme “Kraj” predstavlja završetak rada sa aplikacijom i izlaz iz nje.

**GRESKA**

Nisu uneta sva tri temena trougla.

**GRESKA**

Ne postoji trougao sa zadatim temenima.



