
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: Januar 2015.
Datum: 23.01.2015.

Kandidat:* _____

Broj indeksa:* _____

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

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

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 u okviru (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.

4. [10] Razmatra se sledeći programski fragment:

```
public float foo(int a, int b, float x) {
    if (a>1 && b==0) {          //linija 1
        x=x/a;                  //linija 2
    }
    if (a==2 || x>1) {         //linija 3
        x=x+1;                  //linija 4
    }
    else
        x=x*a;                  //linija 5
    return (a+x);              //linija 6
}
```

a) Nacrtati graf kontrole toka koristeći brojeve linija koda.

b) Odrediti diti ciklomatsku kompleksnot grafa i skup bazičnih putanja.

c) Generisati test primere koji odgovaraju bazičnim putanjama iz tačke b.

d) Izračunati procentualnu pokrivenost iskaza, odluka i elementarnih uslova za testove dobijene u tački c.

5. [10] a) U datom Java kodu, za promenljive date u tabeli, pronaći sve: DU lance i c- i p-upotrebe.

```

1. program Example()
2. var staffDiscount, totalPrice, finalPrice, discount, price
3. staffDiscount = 0.1
4. totalPrice = 0
5. input(price)
6. while(price != -1) do
7.     totalPrice = totalPrice + price
8.     input(price)
9. end_while

10. print("Total price: " + totalPrice)
11. if(totalPrice > 15.00) then
12.     discount = (staffDiscount * totalPrice) + 0.50
13. else
14.     discount = staffDiscount * totalPrice
15. end_if
16. print("Discount: " + discount)
17. finalPrice = totalPrice - discount
18. print("Final price: " + finalPrice)
19. end_program

```

Promenljive	DU lanci	c- upotrebe	p- upotrebe
staffDiscount			
totalPrice			
finalPrice			
discount			
price			

b) Kako će izgledati ovaj program, ako se koristi mutacioni operator SDWD, koji DO-WHILE naredbu zamenjuje sa WHILE? Napisati test primere za koji ovaj mutirani program daje isti, odnosno različiti rezultat u odnosu na početni program.

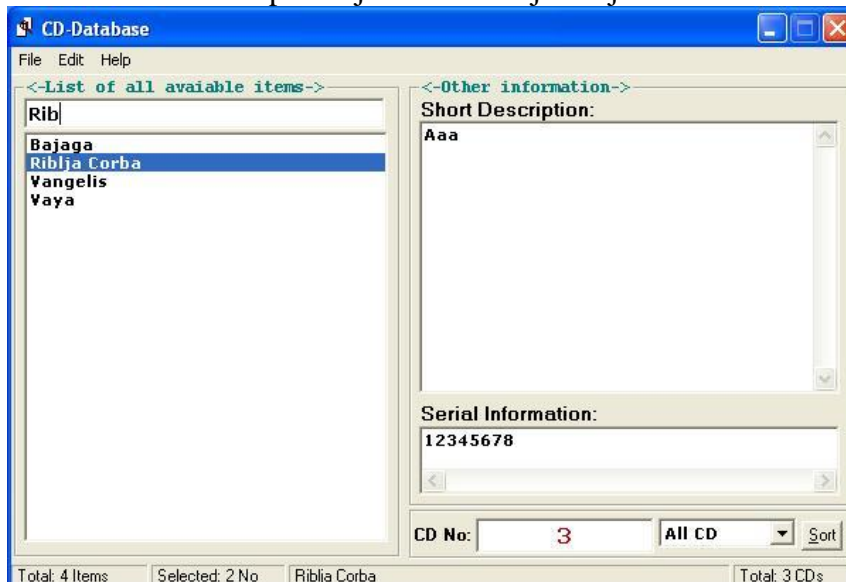
6. [10] Neka su dati sledeći potpisi interfejsa i klasa neke igre na Internetu:

```
interface Telo { ... }
public abstract class Predmet implements Telo { ... }
public class Lopta extends Predmet { ... }
public class Kvadar extends Predmet { ... }
public class Kupa extends Predmet { ... }
public class Valjak extends Predmet { ... }
public class Globus extends Lopta { ... }
public class Zgradica extends Kvadar { ... }
public class Kucica extends Kvadar { ... }
public class Figura implements Telo { ... }
```

- a) Nacrtati u vidu stabla hijerarhiju svih komponenata (klasa/interfejsa) ovog programa i napisati svaki korak postupka integracionog testiranja tehnikama od dna ka vrhu (*bottom-up*) i od vrha ka dnu (*top-down*).
- b) Napisati nedostatke tehnika od dna ka vrhu i od vrha ka dnu.
- c) Ukratko (u jednoj do dve rečenice) objasniti pojam *mock* objekta i dati primer, definisanjem određene gorenavedene klase i potrebnih metoda.

Napomena: Podrazumevati da za svaku komponentu postoji odgovarajuća implementacija i da su komponente povezane, kao što piše u potpisima.

7. [10] Data je sledeća Windows aplikacija za evidenciju svojih diskova:



Korisnik kada startuje aplikaciju vidi listu svih svojih diskova, i može da radi operaciju pretraživanja (unosom podatka u tekstualno polje iznad liste). Selektovanjem određenog diska dobija se kratak opis i serijski broj, kao i redni broj diska u bazi. Klikom na dugme *Sort* (uz odabrano *All Cd* u padajućoj listi) sortira se lista u rastućem abecednom poretku, a ponovnim klikom na dugme *Sort* (uz isto odabrano *All CD*) sortira se lista u opadajućem poretku. Ukoliko je u padajućoj listi odabran neki redni broj diska i klikne se na *Sort*, dobija se samo taj disk u listi sa leve strane. Naziv može imati najmanje 3, a najviše 64 alfanumerička karaktera, serijski broj mora imati tačno 13 cifara, a redni broj diska može imati samo cifre.

U svakom trenutku odabirom opcije *Edit* u gornjem meniju može se odabrati stavka koja ima sledeća polja: tekstualno polje *CD No*, tekstualno polje za naziv diska (*Product Name*), tekstualno polje za kratak opis (*Short Description*), tekstualno polje za serijski broj (*Serial Info*), radio dugmad *Same CD* i *Different CD* i padajuću listu za tip diska (*Music/Movie/Games*). Kada se odabere *Same CD*, u tom slučaju može biti redni broj (*CD No*) kao već neki postojeći disk, a ukoliko se odabere *Different CD*, u tom slučaju redni broj (*CD No*) mora biti jedinstven. Dugmetom *ADD CD* potvrđuje se forma i dodaje disk u listu diskova, a dugmetom *Cancel* otkazuje se dodavanje diska i vraća na prethodni glavni ekran. Ukolikose ne unese neki podatak, korisnik dobija poruku sa greškom.

Kada je selektovan određeni disk u gornjem meniju pod stavkom *Edit* mogu se odabrati: “*Edit record*” i “*Delete record*”. Ažuriranje diska iz baze se radi stavkom “*Edit record*” i tada se mogu izmeniti samo stavke *Short Description* i/ili *Serial Info*, i potvrditi (*Update*) ili odustati (*Cancel*) od izmene. Brisanje diska iz baze se radi stavkom “*Delete record*” i tada korisniku izlazi iskačući prozor, koji javlja pitanje “*Da li ste sigurni?*” i nudi dva dugmete: *YES* i *NO*. U oba slučaja, i ažuriranja, i brisanja, korisnik se vraća na glavni ekran.

a) Metodom podele na klase ekvivalencije napisati test primere kojima biste testirali ulazne i izlazne podatke ove aplikacije.

b) Nacrtați FSM graf (*Finite State machine graph*) za opisanu aplikaciju i prikazati liste sekvenci IP i FIP.

c) Napisati druge tehnike kojima biste testirali datu aplikaciju (i za svaku navesti bar po jedan test primer).