
Elektrotehnički fakultet u Beogradu
Katedra za računarsku tehniku i informatiku

Predmet: Objektno orijentisano programiranje
(OF2OO1, OS2OO1, OS3OOP, OE2OOP, OT2OOP)

Nastavnik: Prof. dr Dragan Milićev

Ispitni rok: Januar 2008.

Datum: 04.02.2008.

Kandidat: _____

Broj Indeksa: _____ *E-mail:* _____

*Ispit ima dva dela ukupnog trajanja 3 sata. Na prvom delu **nije** dozvoljeno korišćenje literature. Na drugom delu **jest**e dozvoljeno korišćenje literature. Trajanje prvog dela student određuje prema ličnom nahođenju. Student dobija drugi deo ispita kad preda prvi deo.*

UPISATI SVOJE PODATKE I NA PRVOJ STRANI DRUGOG DELA ISPITA!

Pismeni ispit:

Zadatak 1 _____/10

Zadatak 2 _____/10

Zadatak 3 _____/10

Zadatak 4 _____/10

Zadatak 5 _____/20

Rad u toku semestra:

Projekat _____/30

Obavezni domaći zadaci _____/10

Neobavezni domaći zadaci _____/10

Ukupno na ispitu: _____/60

Ukupno u toku semestra: _____/50

Ukupno: _____/110

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. Ocenjivanje unutar potpitanja je po sistemu "sve ili ništa", odnosno nema parcijalnih poena. Kod pitanja koja imaju ponudene odgovore treba **samo zaokružiti** jedan odgovor. Na ostala pitanja odgovarati **čitko, kratko i precizno**. Zadaci 1-4 su eliminatorni prema Pravilima predmeta.

1. (10 poena)

Dat je sledeći program na jeziku C++:

```
class Graphic {
public:
    virtual void draw () = 0;
private:
    Graphic () {}
};

class Rectangle : public Graphic {
public:
    Rectangle (int width, int height) : w(width), h(height) {}
private:
    int w, h;
};

void main () {
    Rectangle r1;
    Rectangle r2(4,5);
}
```

Precizno navesti sve greške koje će prevodilac prijaviti prilikom prevođenja ovog programa i označiti linije koda u kojima su te greške.

2. (10 poena)

Zašto ovaj program nije sasvim korektan? Imenovati problem, pokazati gde i zašto on nastaje i kako ga treba rešiti (napisati šta, gde i kako treba izmeniti ili dodati u kodu).

```
#include <iostream.h>
const int N = 100;

class Widget {
public:
    Widget (const char* s, Widget* nxt=0) : string(s), next(nxt) {}
    Widget* getNext () const { return next; }
    const char* getString () const { return string; }
private:
    Widget* next;
    const char* string;
};

void main () {
    Widget* head = 0;
    for (int i=0; i<N; i++) head = new Widget("Hello!\n",head);
    for (Widget* w = head; w!=0; w=w->getNext()) cout<<w->getString();
}
```

3. (10 poena)

Napisati definiciju klase D i definicije svih njenih članova. Klasa D je izvedena iz postojeće klase B i treba da poseduje sledeće članove:

- Privatan podatak član koji je pokazivač na dinamički kreirani objekat postojeće klase X, a koji ekskluzivno pripada objektu klase D.
- Javni podrazumevani konstruktor koji poziva podrazumevani konstruktor osnovne klase i inicijalizuje podatak član tako da ukazuje na novi kreirani dinamički objekat klase X.
- Javni operator dodele koji poziva operator dodele osnovne klase i duboko kopira objekat klase D.
- Javni konstruktor kopije koji poziva konstruktor kopije osnovne klase i duboko kopira objekat dat kao argument.
- Zaštićeni destruktork koji briše pridruženi objekat klase X.

Indeks: _____ Student: _____

4. (10 poena)

Projektuje se sistem za proračunavanje karakteristika zračenja radio antena određenog oblika. *Antena (Aerial)* ima, kao svojstva, visinu i poluprečnik (racionalni brojevi u odgovarajućim jedinicama mere), kao i proračunatu karakteristiku zračenja (niz racionalnih brojeva veličine `MAXSIZE`). Antena ima i operaciju `computeChararcteristics()` koja proračunava karakteristiku date antene sa datim svojstvima i rezultat proračuna smešta u navedeni niz. Proračun karakteristike se može vršiti po više metoda, označenih sa *MethodA*, *MethodB* i *MethodC*. Potrebno je obezbediti mogućnost da se različiti primerci antena mogu konfigurisati tako da se za njih izračunavaju karakteristike različitim metodama, s tim da je skup metoda lako proširiv bez izmene ostatka sistema, a da je konfigurisanje metoda za date primerke antena izvodivo i dinamički (u vreme izvršavanja).

Nacrtati UML dijagram klasa koje modeluju sistem definisan ovim zahtevima i u potpunosti implementirati klasu `Aerial` na jeziku C++. Dati primer nekoliko objekata klase `Aerial` konfigurisanih sa različitim metodama proračuna.

Ako je korišćen neki projektni obrazac, navesti koji i koje klase u ovom sistemu igraju koje apstraktne uloge definisane tim projektnim obrascem.

5. (20 poena) Konstruktivni zadatak

Potrebno je realizovati uprošćenu simulaciju plaćanja na kasama u hipermarketu. Potrošači plaćaju robu na nekoj od N kasa, u trenutku kad dođu na red ispred izabrane kase. U trenutku plaćanja, suma novca kojom potrošač raspolaže biva umanjena za vrednost robe. Odluku na kojoj će kasi čekati potrošač donosi u trenutku kad priđe delu hipermarketa sa kasama, prema trenutnom broju potrošača u svakom od redova ispred kasa. Po želji, koristiti i/ili menjati postojeći programski kod TSS. Po završenoj simulaciji, potrebno je ispisati sledeće rezultate: zbirni promet za svaku kasu, broj potrošača koji su prošli kroz svaki od redova ispred kase i prosečno zadržavanje potrošača za svaki od redova.

- a) (10 poena) Napisati klase potrebne za opisanu simulaciju. Eventualne promene u TSS jasno navesti.
- b) (5 poena) Dodati u postojeće **visitor** klase metode odgovarajuće novim klasama.
- c) (5 poena) Napisati programski kod koji stvara i ispravno povezuje odgovarajući broj primeraka opisanih klasa, te pokreće simulaciju i ispisuje tražene rezultate simulacije. Navesti klase koje su eventualno potrebne za kompletnost simulacije, ukratko ih opisati (samo imena i opis metoda, bez implementacije) i stvoriti odgovarajući broj njihovih primeraka. Na kraju, osloboditi korišćenu dinamičku memoriju.