

Objektno-orijentisano programiranje, ispit JUN1
Matematički fakultet, školska godina 2024/2025

Uputstvo: Na Desktop-u raspakovati zip arhivu `oop.i.zip` (desni klik → Extract Arhieve Here...). Preimenovati projekat svojim podacima (npr. `oop.i_IA_MarkoMarkovic_mi23101`). Pokrenuti *IntelliJ Idea* i otvoriti projekat sa Desktop-a (putanja: `oop/Desktop/oop.i...`). U kreiranim paketima `zadatak1`, `zadatak2` i `zadatak3` pisati rešenja zadataka. Paketi se nalaze na putanji `src/main/java/`.

Kôd **ne sme** imati sintakasnih grešaka niti izbacivanje `NullPointerException`-a.

U tekstu je dat opis klase, njihovih atributa i metoda. **Dozvoljeno** je dodati nove attribute, klase, metode, enumere, interfejsu u slučaju da olakšavaju implementaciju i/ili poboljšavaju kvalitet koda i slično.

Da bi se uspešno položio ispit potrebno je osvojiti **barem 30 poena**.

1. [20p] Agencije prate stanje na tržištu nekretnina i na taj način pomažu klijentima da pronađu nekretninu prema odgovarajućim kriterijumima.
 - Definisati klasu `Nekretnina` sa atributima `String adresa`, `double površina` i `double cena`. U klasi definisati konstruktor i apstraktnu metodu `String opis()`.
 - Definisati nabrojivi tip `Tip`, koji sadrži tip nekretnine: prodaja ili iznajmljivanje. Definisati statičku metodu `izStringa(String s)` koja vraća odgovarajući tip na osnovu argumenta.
 - Definisati klasu `Stan` koja nasleđuje klasu `Nekretnina`, sa atributima `int brojSoba` i `Tip tip`. Dodati konstruktor i implementirati metodu `opis` tako da vraća informacije o stanu u obliku:
`Stan: adresa, N soba, površina P m2, cena X EUR (tip)`
 - Definisati klasu `Agencija` koja čuva listu nekretnina. U klasi implementirati:
 - konstruktor bez argumenata,
 - metodu `dodaj(Nekretnina n)` koja dodaje novu nekretninu u listu,
 - metodu `ispisiSve()` koja ispisuje sve nekretnine u listi,
 - metodu `filtrirajPoTipu(String tip)` koja vraća listu stanova određenog tipa.
 - Definisati klasu `Test` u kojoj treba napraviti po jedan objekat tipa `Agencija` u koji zatim treba dodati nekoliko stanova.
 - Ispisati sve nekretnine pomoću metode `ispisiSve`.
 - Prikazati samo stanove za prodaju i samo stanove za iznajmljivanje koristeći metodu `filtrirajPoTipu`.
2. [20p] Implementirati aplikaciju u klasi `RaspodelaZaDomove` korišćenjem `JavaFX` biblioteke koja izgleda kao na slici
1. U datotekama `matf.txt`, `etf.txt`, `farm.txt` i `medf.txt` date su informacije o studentima koji su dobili dom. Dimenzije prozora su 400x400.
 - Klikom na dugme `Prikazi` ispisuju se informacije o studentima koji su dobili dom na označenom fakultetu i označenog pola. U slučaju da nije označeno polje za fakultet ili pol, ispisati poruku o grešci u labeli za greške crvenom bojom. U zavisnosti od označenog fakulteta, čitaju se informacije iz datoteke koja sadrži informacije o studentima tog fakulteta. Obezbediti da se učitavanje iz datoteke izvrši tačno jednom.

Sadržaj datoteke `matf.txt`:

```
DEVOJKE
Ana Markovic, 240 espb, 9.98
Milena Simic, 180 espb, 10.0
Tina Radovic, 180 espb, 9.67
Nikolina Konatar, 120 espb, 9.80
Milica Scepanovic, 120 espb, 8.95
...

MOMCI
Marko Markovic, 240 espb, 10.0
Lazar Minic, 240 espb, 9.98
Nemanja Nikolic, 180 espb, 10.0
Petar Lazic, 240 espb, 9.80
Milos Radovic, 180 espb, 9.67
...
```

(a) Inicijalni prozor

(b) Prikaz u slučaju da pol nije označen

Ana Markovic,	240 espb,	9.98
Milena Simic,	180 espb,	10.0
Tina Radovic,	180 espb,	9.67
Nikolina Konatar,	120 espb,	9.80
Milica Scepanovic,	120 espb,	8.95
Katarina Markovic,	110 espb,	9.05
Ana Nedic,	60 espb,	9.0
Marija Onic,	60 espb,	8.80
Ana Radovic,	40 espb,	7.05

(c) Prikaz ukoliko je korisnik označio i fakultet i pol

Slika 1: Raspodela za domove

3. [25p] U klasi **Stek** implementirati generičku strukturu podataka stek koja može čuvati elemente proizvoljnog tipa.

- Definirati metodu `boolean prazan()` koja vraća `true` ako je stek prazan, u suprotnom `false`.
- Definirati metodu `int velicina()` koja vraća broj elemenata u steku.
- Definirati metodu `void ubaci(T element)` koja dodaje element na vrh steka.
- Definirati metodu `T ukloni()` koja uklanja i vraća element sa vrha steka. Ukoliko je stek prazan, izbaciti izuzetak tipa `StekJePrazan`.
- Definirati metodu `T vidiVrh()` koja vraća element sa vrha steka bez uklanjanja. Ukoliko je stek prazan, izbaciti izuzetak tipa `StekJePrazan`.
- Definirati statičku metodu `int porediVrh(Stek<T> s1, Stek<T> s2)` koja poredi elemente na vrhu 2 steka. Metoda vraća 0 ukoliko su elementi na vrhu jednaki, negativnu vrednost ukoliko je element prvog steka manji od elementa drugog, u suprotnom se vraća pozitivna vrednost.
- Definirati klasu `Test` koja testira implementirane metode za rad sa stekom čiji su elementi tipa `Krug`. `Krug` u ravni ima atribut za poluprečnik i centralnu tačku. `Krug k1` je manji od kruga `k2` ukoliko ima manji poluprečnik. Krugovi su jednaki ukoliko imaju isti poluprečnik i centar. Koristiti metodu `toString()` za ispisivanje objekata tipa `Stek`, `Krug` i `Tacka`.

Napomena: Maksimalan broj poena u slučaju korišćenja ugrađenih struktura podataka je 12.