

## Objektno-orijentisano programiranje, Septembar 1

Matematički fakultet, školska godina 2021/2022

**Napomena:** Na Desktop-u napraviti direktorijum pod imenom `oop_Asistent_Prezime_Ime_Indeks` (npr. `oop_NM_Peric_Pera_mi12082`). Pokrenuti *Intellij Idea* i u napravljenom direktorijumu napraviti projekat sa istim nazivom. U napravljenom projektu, paket takođe nazvati tako.

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

Vreme za rad: **2.5 sata**

Inicijali: IR: 2I1A — SS: 2I1B — VK: 2I2A, 2I2B

U tekstu je dat opis klasa, njihovih atributa i metoda. **Dozvoljeno** je (i ohrabrujemo Vas) dodati nove attribute, klase, metode, enume, interfejsu u slučaju da Vam olakšavaju implementaciju, i/ili smatrate da Vam poboljšavaju kvalitet koda i slično. Nekada će zahtevi u zadatku i zahtevati od Vas da dodate novi atribut ili slično.

Da bi se uspešno položio ispit potrebno je osvojiti **barem 25 poena**.

1. **2 poena** Napraviti marker interfejs `Povlasceni` koji predstavlja nosioce na žrebu za UEFA takmičenja.
2. **4 poena** Napraviti apstraktnu klasu `Tim` koja sadrži ime fudbalskog kluba (`String`) kao i njegov koeficijent na UEFA rang listi (`double`). Koeficijent se izračunava kao prosek osvojenih bodova u UEFA takmičenjima u poslednjih 5 sezona. Napisati konstruktor koji dobija listu bodova osvojenih u poslednjih 5 sezona (`List<double>`) i ime kluba. Implementirati potrebne `get` i `set` metode. Ova klasa takođe treba da sadrži i apstraktni metod `double verovatnocaOsvajanja` koji govori o tome koliku verovatnoću tim ima da osvoji takmičenje. Napisati metod `toString()` koji vraća `String` reprezentaciju tima u formatu `Ime koeficijent verovatnocaOsvajanja` kao na slici.
3. **3 poena** Napraviti klasu `Nosilac` koja nasleđuje klasu `Tim` i implementira interfejs `Povlasceni`. Napisati odgovarajući konstruktor za ovu klasu. Verovatnoća osvajanja takmičenja za nosioce se računa kao njihov koeficijent pomnožen sa 30 i podeljen sa 100.
4. **3 poena** Napraviti klasu `Nenosilac` koja nasleđuje klasu `Tim`. Ova klasa se dodatno karakteriše i podatkom `int nacionalniKoeficijent`. Napisati konstruktor za ovu klasu koji pored neophodnih podataka dobija i vrednost koja predstavlja nacionalni koeficijent. Verovatnoća osvajanja takmičenja za nenosioce se računa kao njihov koeficijent pomnožen sa nacionalnim koeficijentom i podeljen sa 100.
5. **5 poena** Napraviti klasu `Grupa` koja predstavlja grupu u jednom takmičenju. Od atributa ova klasa ima `String nazivGrupe` i listu timova koji se takmiče (`List<Tim>`). Napisati potrebne `get` i `set` metode. Dodatno, definisati metodu `void dodajTim(Tim tim)` koja dodaje jedan tim u grupu. Definirati metod `toString` koji ispisuje naziv grupe, a zatim i spisak svih timova iz te grupe. Pogledati primer.
6. **7 poena** Napraviti apstraktnu klasu `UEFATakmicenje` koja će predstavljati jedno fudbalsko takmicenje pod okriljem UEFA. Ova klasa sadrži ime takmičenja, listu timova koji se takmiče ove sezone (podatak tipa `List<Tim>`) i listu grupa koje postoje u takmičenju (`List<Grupa>`). Takođe, klasa treba da ima atribut `koeficijentTakmicenja` koji je ceo broj i predstavlja jačinu konkretnog takmičenja, odnosno njegov koeficijent. Napisati:
  - Konstruktor koji dobija naziv takmičenja, koeficijent, listu timova koji se takmiče te godine i grupe.
  - Metod `double nagradniFond()` koji izračunava nagradni fond takmičenja. Nagradni fond se računa po formuli  $\text{prosecniKoeficijentTimova} * \text{koeficijentTakmicenja} / \text{brojTimova}$ .
  - Apstraktni metod `double jacinaTakmicenja()` koji računa jačinu takmičenja za jednu sezonu na osnovu jačine učesnika te sezone.
  - metod `String prikaziGrupe()` koji ispisuje sve grupe u takmičenju kao na slici.
  - metod za prikaz svih timova koji se takmiče te godine u takmičenju.
7. **2 poena** Napraviti klasu `LigaSampiona` koja nasleđuje klasu `UEFATakmicenje`. Jačina Lige Šampiona se računa kao prosečni koeficijent svih njenih učesnika (svih timova koji se takmiče te sezone).
8. **2 poena** Napraviti klasu `LigaEvrope` koja nasleđuje klasu `UEFATakmicenje`. Jačina Lige Evrope se računa kao prosek najjačeg i najslabijeg tima po koeficijentu u takmičenju.
9. **4 poena** Napraviti klasu `LigaKonferencija` koja nasleđuje klasu `UEFATakmicenje`. Jačina Lige Konferencija se računa tako što se isključi 5 najjačih i 5 najslabijih timova po koeficijentu i izračuna se prosek preostalih timova.
10. **3 poena** Napraviti šablonsku klasu `TezinaZreba` koja se odnosi samo povlašćene timove na žrebu. Implementirati metod `double tezinaZreba()` koji vraća težinu žreba za povlašćeni tim. Težina žreba za jedan tim se računa kao prosečni koeficijent timova koji nisu povlašćeni a u istoj su grupi.

11. **25 poena** Napraviti klasu UEFA koja nasleđuje klasu Application biblioteke javafx i izgleda kao na slikama.

Iz datoteke takmicenje.txt se učitavaju podaci o takmičenju i učesnicima. U prvom redu datoteke se nalazi šifra takmičenja, a zatim i koeficijent takmičenja. U narednim linijama se nalaze podaci o timovima. Za svaki tim postoji njegovo ime, zatim 5 vrednosti koje predstavljaju bodove osvojene u poslednjih 5 godina. Nakon bodova se nalazi podatak o tome da li je tim povlašćen na žrebu ili ne. Za nepovlašćene ekipe postoji i podatak o nacionalnom koeficijentu. Poslednji podatak u je naziv grupe u kojoj se tim nalazi.

Šifra takmičenja određuje njegov naziv na sledeći način:

- LS - Liga Sampiona
- LE - Liga Evrope
- LK - Liga Konferencija

Klikom na dugme ucitaj se učitavaju svi podaci iz datoteke. Moguće je izlistati sve timove u konkretnom takmičenju ili sve grupe.

Klikom na dugme Grupe se izlistavaju grupe sortirane po nazivu rastuće. Za svaki povlašćeni tim se dodatno ispisuje i težina njegovog žreba. Videti primer.

Klikom na dugme Timovi se izlistavaju svi timovi u takmičenju i to na sledeći način:

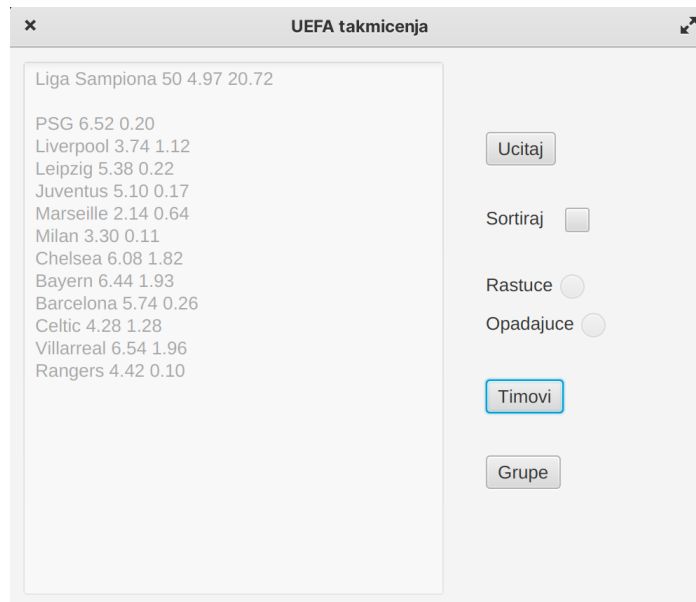
- Ako je čekirano polje Sortiraj, timovi se sortiraju rastuće ili opadajuće prema nazivu, zavisno od toga da li je čekirano radio dugme Rastuce ili Opadajuće.
- Ako polje Sortiraj nije čekirano, timovi se izlistavaju u slučajnom/izmešanom poretku (Ne redom kojim su učitani).

Omogućiti da se ne može kliknuti nijedno radio dugme ukoliko dugme Sortiraj nije čekirano.

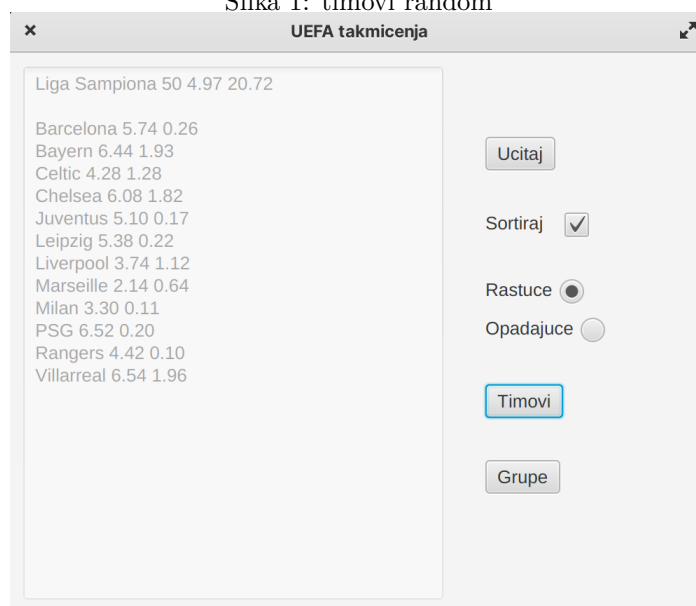
12. Primer sadržaja datoteke:

```
LS 50
Chelsea 2.4 6.5 9.3 10 2.2 Povlasceni B
Milan 1.4 4.3 7.2 1.1 2.5 Nepovlasceni 3.4 A
Barcelona 6.5 3.9 8.4 1.7 8.2 Nepovlasceni 4.5 C
Liverpool 2.5 2.5 3.4 4.6 5.7 Povlasceni A
Celtic 1.2 3.4 4.5 5.6 6.7 Povlasceni C
Rangers 1.7 8.2 3.5 5.1 3.6 Nepovlasceni 2.3 B
Bayern 9.1 10 3.4 8.2 1.5 Povlasceni C
PSG 9.3 4.5 2.4 7.5 8.9 Nepovlasceni 3.0 B
Juventus 1.3 5.2 6.7 8.7 3.6 Nepovlasceni 3.4 A
Villarreal 9.8 3.4 5.2 7.5 6.8 Povlasceni D
Leipzig 2.4 5.7 1.4 9.6 7.8 Nepovlasceni 4.0 D
Marseille 1.1 2.3 1.3 4.1 1.9 Povlasceni D
```

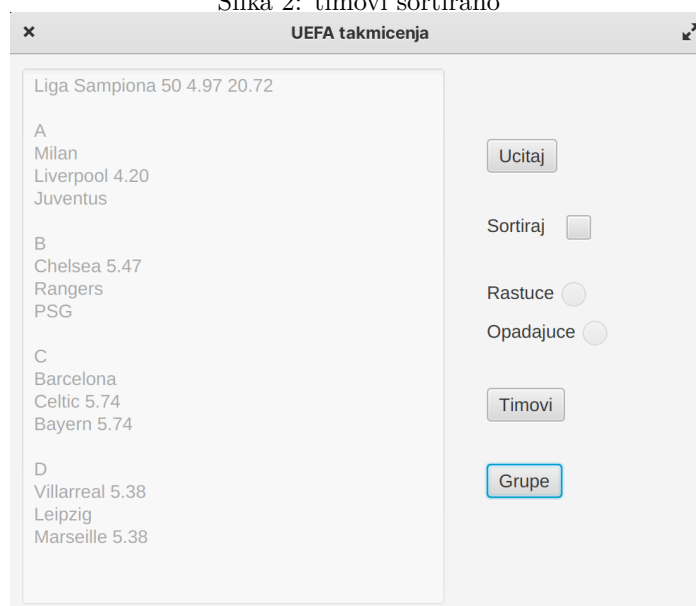
13.



Slika 1: timovi random



Slika 2: timovi sortirano



Slika 3: grupe