

Objektno-orientisano programiranje, Septembar 1

Matematički fakultet, školska godina 2019/2020

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: **3 sata**

Inicijali: BS: 2MNVA, 2RL1A, 2RL1B | NM: 2RL2A, 2RL2B | AB: 2I1A, 2I2A, 2MNVB | DA: 2I1B, 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. Napraviti apstraktnu klasu `Listic` koja se karakteriše poljem `premija` (`int`). Implementirati konstruktor koji prima vrednost za premiju i `get` metod za to polje. Klasa sadrži apstraktne metode `int odrediDobitak()` koji izračunava i vraća vrednost nagrade i `String proveriDobitak()` koji proverava da li je učesnik osvojio neku nagradu i vraća odgovarajuću poruku. Implementirati i statički metod `List<Integer> izvlacenje(int brojLoptica, int gornjaGranica)` koji izvlači `brojLoptica` brojeva iz intervala `[1, gornjaGranica]`. Brojevi moraju biti jedinstveni, odnosno lista ne sme sadržati duplikate.
2. Napraviti klasu `Loto` koja nasleđuje klasu `Listic` i predstavlja jednu kombinaciju brojeva koje je učesnik zaokružio na listiću. Klasa se karakteriše poljem `lotoListic` (`List<Integer>`, brojevi koji su zaokruženi na listiću) i statičkim poljem `dobitnaKombinacija` (`List<Integer>`). Implementirati:
 - konstruktor koji prima vrednost za premiju i listić;
 - potrebne `get` i `set` metode;
 - metod `int odrediBrojPogodaka()` koji računa koliko se brojeva zaokruženih na listiću nalazi u dobitnoj kombinaciji;
 - apstraktni metod `int odrediDobitak()` koji računa vrednost nagrade na osnovu broja pogodaka. Ukoliko je broj pogodaka manji od 3, nema nagrade. Ukoliko je pogodena dobitna kombinacija, nagrada je cela premija, a za svaki pogodak manje premija je manja deset puta;
 - apstraktni metod `String proveriDobitak()` koji na osnovu vrednosti nagrade vraća odgovarajuću poruku: `Nema dobitka!` ukoliko nema nagrade, inače poruku sa informacijama o broju pogodaka i veličini nagrade (kao na slici 1);
 - metod `toString` koji formira i vraća nisku sačinjenu od odabranih brojeva sa listića (kao na slici 1).
3. Napraviti klasu `Bingo` koja nasleđuje klasu `Listic` i predstavlja jedan Bingo listić. Klasa se karakteriše poljem `bingoListic` (`int[][]`, matrica brojeva dimenzije 3x5) i statičkim poljem `izvuceniBrojevi` (`List<Integer>`). Svaka kolona matrice odgovara po jednom od slova BINGO, redom. Implementirati:
 - konstruktor koji prima vrednost za premiju i listić;
 - potrebne `get` i `set` metode;
 - metod `boolean postignutBingo()` koji određuje da li je za svako slovo reči BINGO izvučen makar jedan broj iz odgovarajuće kolone.
 - apstraktni metod `int odrediDobitak()` koji računa vrednost nagrade u zavisnosti od toga da li je postignut bingo. Ukoliko je postignut bingo, nagrada je cela premija, u suprotnom učesnik ne dobija ništa;
 - apstraktni metod `String proveriDobitak()` koji u slučaju da je postignut bingo vraća nisku sa informacijom o osvojenoj nagradi, a u suprotnom poruku da nema dobitka;
 - metod `toString` koji formira i vraća nisku od reči BINGO u prvom redu, a zatim vrednostima iz matrice (kao na slici 1).
4. Napraviti generičku klasu `IgreNaSrecu<T>` koja se karakteriše poljem `igre` (`List<List<T>>`). Jedan element liste predstavlja listu igara koje je odigrao jedan učesnik. Implementirati:
 - potrebne `get` i `set` metode;
 - metod `String proveriSrecu(T igra)` koji za igru tipa `Listic` vraća nisku dobijenu pozivom `proveriDobitak`, a u suprotnom vraća poruku da nema dobitka;
 - metod `void dodajIgru(boolean noviUcesnik, T igra)` koji dodaje učesniku igru u listu igara. Ukoliko je vrednost `noviUcesnik` jednaka `true` potrebno je napraviti novu listu za tog učesnika, inače treba dodati igru u poslednju listu liste `igre`;

- metod `toString` koji formira nisku tako da za svakog učesnika prvo ispiše njegov redni broj, a zatim sve igre koje je odabrao uređene prema tipu tako da se prvo ispisuju Loto igre, zatim Bingo igre, a ostale na kraju. Za svaku od igara ispisati poruku sa informacijama o dobitku.

5. Napraviti klasu `Lutrija` koja nasleđuje `Application` klasu biblioteke `javafx` i izgleda kao na slikama. Klasa sadrži jedno statičko polje `igreNaSrecu` (`igreNaSrecu<Listic>`).

Klikom na dugme `Ucitaj` iz datoteke `lutrija.txt` učitavaju se informacije o igrama i smeštaju u listu `igre` polja `igreNaSrecu`. Primer datoteke dat je na kraju teksta.

Klikom na dugme `Izvlacenje` izvlače se brojevi za Loto (7 brojeva ne većih od 39) i Bingo (15 brojeva ne većih od 75). Obezbediti da se izvlačenje brojeva može izvršiti najviše jednom tokom rada programa.

Klikom na dugme `Proveri srecu` za svakog igrača ispisuju se redni broj, listići i odgovarajuće poruke. Obezbediti da se ova akcija ne može izvršiti pre izvlačenja brojeva.

Klikom na dugme `Najveci dobitak` ispisuje se najveći od svih dobitaka koji je neko od učesnika osvojio.

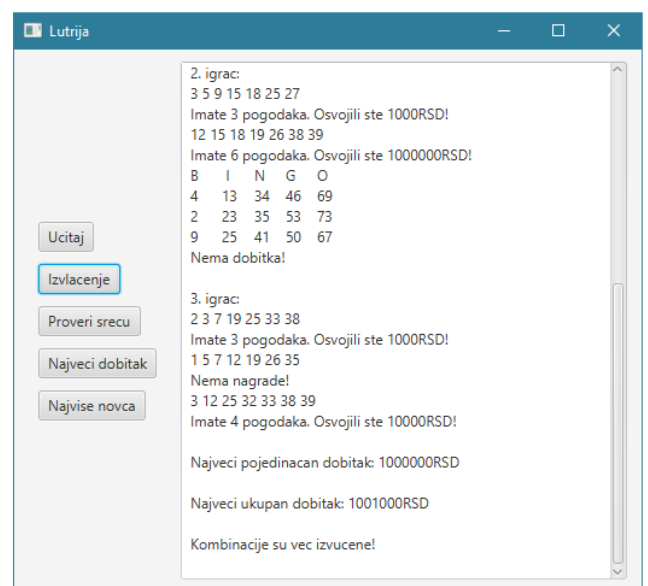
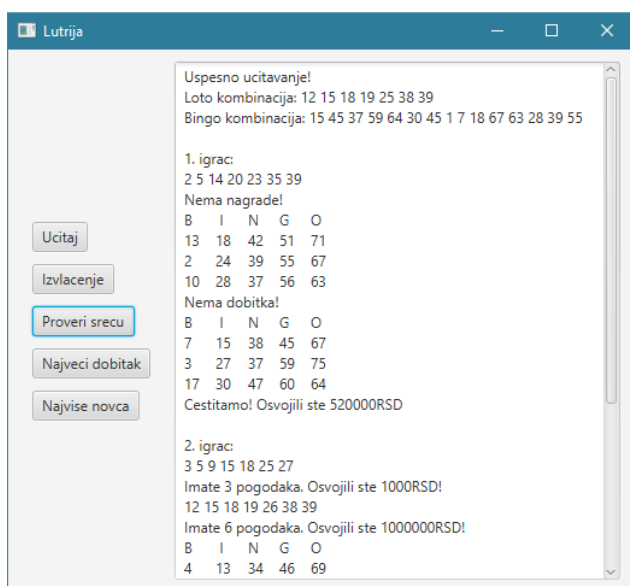
Klikom na dugme `Najvise novca` ispisuje se najveći ukupni dobitak koji je neko od učesnika osvojio.

Dozvoljeno je proširiti klase dodatnim atributima i metodama kako biste realizovali prethodno navedene zahteve.

Prvi red datoteke `lutrija.txt` je oblika: `premijaLoto, premijaBingo`. Nakon toga sledi proizvoljan broj učesnika i njihovih igara, sve u posebnom redu. Sve igre predstavljene su listom celim brojevima razmaknutih blanko karakterom. Igre učesnika razdvojene su redom koji sadrži zvezdice.

Učesnika može biti proizvoljno i svako može odigrati proizvoljan broj igara. Primer sadržaja datoteke:

```
10000000, 520000
*****
13 18 42 51 71 2 24 39 55 67 10 28 37 56 63
2 5 14 20 23 35 39
7 15 38 45 67 3 27 37 59 75 17 30 47 60 64
*****
12 15 18 19 26 38 39
4 13 34 46 69 2 23 35 53 73 9 25 41 50 67
3 5 9 15 18 25 27
*****
3 12 25 32 33 38 39
1 5 7 12 19 26 35
2 3 7 19 25 33 38
```



Slika 1: Klik na Učitavanje, Izvlačenje i Provera sreće (nastavak ispisa je na drugoj slici).

Slika 2: Klik na Najveći dobitak, Najviše novca i ponovo Izvlačenje.