

Objektno programiranje, Kolokvijum, Grupa 1

Matematički fakultet

Školska godina 2017/2018

Napomena: Napraviti direktorijum /home/ispit/IdeaProjects. Pokrenuti *Intellij Idea* i u napravljenom direktorijumu napraviti projekat pod imenom oop_Ime_Prezime_Indeks_Asistent (npr. oop_Pera_Peric_mi12082_NM). U napravljenom projektu, paket takođe nazvati isto tako.

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

Vreme za rad: **1.5 sat**

Minimalan potreban broj poena da bi se položio kolokvijum: **5 poena**

Potrebno je napisati barem jednu klasu i testirati njen rad u test klasi.

Inicijalni asistenti: Biljana - BS, Anja - AB, Božidar - BA, Nemanja - NM

1. (5 poena) Napraviti klasu `TarotKarta` koju karakterišu atributi `vreme` (`int`), `opis` (`String`) i `poreklo` (`String`). Atribut `vreme` može imati tri vrednosti, -1: prošlost, 0: sadašnjost, 1: budućnost.

Implementirati:

- Konstruktor koji prihvata vrednosti za sva tri atributa.
- Konstruktor kopije.
- `get` metode za atribute.
- Metod `String toString()` (pogledati primere ispod).

Ukoliko se konstruktoru prosledi vrednost za `vreme` koja nije -1, 0 ili 1, postaviti je na 0 - odnosno sadašnjost.

```
[tarot karta] vreme: sadasnjost, opis: Visoka Svestenica, poreklo: Italija
```

```
[tarot karta] vreme: buducnost, opis: Lazov, poreklo: Otomansko Carstvo
```

```
[tarot karta] vreme: proslost, opis: Mag, poreklo: Nemacka
```

Napraviti klasu `TestTarotKarta` koja instancira tri navedene Tarot karte i ispisuje ih na standardni izlaz.

2. (5 poena) *[wikipedia]* `Prorok` - osoba za koju se veruje da je u vezi sa *natprirodnom silom ili božanstvom*, i koja služi kao posrednik za otkrivanje određenih znanja ljudima.

Napraviti baznu klasu `Prorok` koja ima atribute `ime` (`String`), broj ostvarenih proročanstava (`int`) i ukupan broj proročanstava (`int`). Obezbediti:

- Konstruktor koji prihvata sva tri atributa.
- Konstruktor kopije.
- `get` metode za sve atribute.
- Metod `void zabeleziIshodProrocanstva(boolean status)` koji ažurira broj uspehlih proročanstva (proročanstvo je uspelo ako je `status true`) i ukupan broj proročanstava.

3. (6 poena) Napraviti klasu `KuglaProrok` (Slika ??) koji nasleđuje klasu `Prorok` i opisuje proroka koji svoja proročanstva izvodi na osnovu kristalne kugle. Klasa poseduje atribut tip kristala (`String`) koji opisuje tip kristala od kojeg je sačinjena kristalna kugla koju prorok koristi. Tip kristala može biti:

- monociklični
- tetragonalni
- kineski

Klasa sadrži statički niz stringova koje predstavljaju moguće proročanstvo `reciProrocanstvo` koji ima vrednosti:

```
["Godine 2020. bice 2020. godina!", "Sledece godine iPhone ce biti jos skuplji.",  
"Sutra ce biti vrucina.", "Godine 2018. na kolokvijumu iz OOP-a nece biti polimorfizma."]
```

Obezbediti:

- Konstruktor koji dodatno ima i tip kristala koji se prosleđuje. Ako nije prosleđen adekvatan tip kristala, postaviti ga na `kineski`.
- `get` metod za tip kristala.
- `String proreknj()` koji ciklično vraća proročanstva iz niza `reciProrocanstvo` počevši od nultog elementa.
- `String toString()` koji ispisuje proroka sa kuglom u obliku datom test primerom ispod.

Primer za metod `toString` (`ime`, `uspela proročanstva`, `ukupno proročanstava`, `tip kugle`).

```
[kugla prorok]: VidovitaZorka, uspelo: 35, ukupno: 42, tip kristala: heksagonalni
```

4. (6 poena) Napraviti klasu `TarotProrok` (slika ??) koji nasleđuje klasu `Prorok` i opisuje proroka koji svoja proročanstva izvodi na osnovu tarot karata. Klasa poseduje atribut karte (`TarotKarta[]`) koji predstavlja niz tarot karata koje prorok poseduje.

Obezbediti:

- Konstruktor koji dodatno ima i niz karata koje se prosleđuju.
- `get` metod za karte.
- `TarotKarta izvuciKartu()` koji vraća Tarot kartu ciklično iz niza počevši od indeksa 0.
- `String proreci(TarotKarta tk)` koji vraća string koji predstavlja ishod pročitane karte (karta se čita tako što se prvo dekodira vreme a potom se dopiše opis karte - videti test primer na samom kraju zadatka)
- `String toString()` koji ispisuje tarot proroka u obliku datom test primerom ispod.

Primer za metod `toString` (ime, uspela proročanstva, ukupno proročanstava, broj karata u špilu).

```
[tarot prorok]: MilanTarot, uspelo: 2, ukupno: 117, poseduje karata: 30
```

5. (3 poena) Napraviti klasu `KucaProrocanstava` koja testira prethodno implementirane klase odnosno instancirati dva proroka, jednog sa tarot kartama, drugog sa kristalnom kuglom.

Sa standardnog ulaza se prvo unose podaci za proroka sa tarot kartama. Unosi se ime, broj uspehlih proročanstava, ukupan broj proročanstava, a za tarot karte uzeti sledeće karte (ručno ih instancirati):

```
[tarot karta] vreme: sadasnjost, opis: Visoka Svestenica, poreklo: Italija  
[tarot karta] vreme: buducnost, opis: Lazov, poreklo: Otomansko Carstvo  
[tarot karta] vreme: prosllost, opis: Mag, poreklo: Nemacka
```

Nakon toga se unose podaci za proroka sa kuglom. Unosi se ime, broj uspehlih proročanstava, ukupan broj proročanstava i tip kugle.

Na standarni izlaz ispisati najpre procenat ostvarenih proročanstava za oba proroka. Potom generisati nova proročanstva, odnosno pozvati metod `proreci` i na standardni izlaz ispisati njihovo tumačenje.

Unesite ime proroka (jedna rec):

MilanTarot

Unesite broj ostvarenih prorocanstava:

2

Unesite ukupan broj prorocanstava:

117

Unesite ime proroka (jedna rec):

VidovitaZorka

Unesite broj ostvarenih prorocanstava:

35

Unesite ukupan broj prorocanstava:

42

Unesite tip kristala (jedna rec)

monociklicni

MilanTarot - procenat uspesnosti: 1.7094018

VidovitaZorka - procenat uspesnosti: 83.33333

MilanTarot prorice: sadasnjost Visoka Svestenica

VidovitaZorka prorice: Godine 2020. bice 2020. godina!



Slika 1: Prorok sa kristalnom kuglom



Slika 2: Prorok sa tarot kartama