

Objektno-orijentisano programiranje, Kolokvijum, Grupa 2

Matematički fakultet

Školska godina 2018/2019

Napomena: Na Desktop-u napraviti direktorijum pod imenom `oop_Ime_Prezime_Indeks_Asistent` (npr. `oop_Pera_Peric_mi12082_NM`). Pokrenuti *Intellij Idea* i u napravljenom direktorijumu napraviti projekat sa istim nazivom. 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, Ivan - IR, Nemanja - NM, Rastko - RD

- (2 poena) Napraviti interfejs `Blokira` koji se sastoji iz metoda `boolean isBlokira()` i `void setBlokira(boolean blok)`. Metodi funkcionišu kao `get` i `set` metode.
- (4 poena) Napraviti baznu klasu `Entitet` (implementira interfejs `Blokira`) koju karakterišu atributi `x` (`double`), `y` (`double`), `vitalnost` (`int`) i `blokira` (`boolean`). Implementirati:

- Konstruktor bez argumenata koji postavlja vrednosti svih numeričkih polja na 0 a polje `blokira` na `true`.
- Konstruktor koji prihvata vrednosti za prva tri polja, a vrednost polja `blokira` postavlja na `true`. Ako se prosledi negativna vrednost za polje `vitalnost`, postaviti polje `vitalnost` na 0.
- `get` metode za sve attribute.
- Metod `toString()` koji vraća nisku kao u primeru ispod.

```
// format ispisa: "Entitet: (<x>, <y>) [<vitalnost> HP] <blokira: + ako je true, - ako je false>"
Entitet: (12.3, 22.4) [100 HP] +
Entitet: (100.345, 12) [0 HP] -
```

- (5 poena) Napraviti klasu `PokretniEntitet` koja nasleđuje klasu `Entitet` i koja dodatno sadrži polje `brzina` (`double`) i polje `ugao` (`double`, `ugao` izražen u radijanima). Implementirati:
- Konstruktor koji prihvata vrednosti za oba atributa i poziva konstruktor bazne klase.
 - `get` i `set` metode za attribute `brzina` i `ugao`.
 - Metod `void pomeriSe()` - pomera entitet u smeru `ugao` za vrednost polja `brzina`:

$$x_{novo} = x_{staro} + brzina \cdot \cos(ugao)$$

$$y_{novo} = y_{staro} + brzina \cdot \sin(ugao)$$

- Metod `toString()` koji vraća nisku kao u primeru ispod (koristiti `toString()` metod natklase).

Napraviti klasu `TestEntitet` i tu instancirati jedan `PokretniEntitet` po uzoru na primer iznad i ispisati ga na standardni izlaz. Za `ugao` uzeti vrednost $\pi/4$ (ignorirati potencijalne razlike u zaokruživanju).

```
PokretniEntitet: (20.0, 12.0) [50 HP] + | brzina: 10.0, ugao: 0.785398...
```

- (4 poena) Napraviti klasu `DeoOpreme` koja takođe nasleđuje klasu `Entitet` i koja se dodatno karakteriše poljima `ime` (tipa `String`) i `cena` (tipa `int`). Implementirati:
- Brojanje napravljenih objekata ove klase.
 - Konstruktor koji prihvata vrednosti za oba atributa i poziva konstruktor bazne klase.
 - Konstruktor kopije.
 - `get` metode za attribute `ime` i `cena`.
 - Metod `toString()` koji vraća nisku kao u primeru:

U klasi `TestEntitet` instancirati objekat tipa `DeoOpreme` po uzoru na primer ispod i ispisati ga na standardni izlaz.

```
DeoOpreme: Veliki stit (1550 zлата)
```

5. (6 poena) Napraviti klasu `Igrac` koja nasleđuje klasu `PokretniEntitet` koja se dodatno karakteriše poljem `ime` (tipa `String`), poljem `inventar` (tipa niz `DeoOpreme`) i poljem `zlato` (tipa `int`). Implementirati:

- Konstruktor koji prihvata podatke o imenu, poziciji, polje `inventar` inicijalizuje na prazan niz kapaciteta 5, polje `brzina` inicijalizuje na 1, a polje `ugao` na 0. Svakom igraču se daje inicijalno 10000 zлата i 100 HP-a.
- `get` metod za atribute `ime` i `zlato`.
- Metod `void kupi(DeoOpreme artikal)` - kupuje `artikal` po njegovoj ceni - smanjuje se polje `zlato` za vrednost cene. Kupljeni `artikal` se dodaje u niz `inventar`. Igrač ne može da kupi više delova opreme od kapaciteta niza `inventar` ili ukoliko nema dovoljno zлата (ne raditi ništa u tim slučajevima).
- Metod `toString()` (pogledati primere ispod).

U klasi `TestEntitet` instancirati objekat tipa `Igrac` i objekat tipa `PokretniEntitet` po uzoru na primere ispod, staviti ih oba u niz tipa `PokretniEntitet[]` i polimorfno pozvati metod `pomeriSe()` nad objektima niza (u petlji).

```
// ispis pre pomeranja:
PokretniEntitet: (0.0, 0.0) [100 HP] + | brzina: 1.0, ugao: 0.0
Igrac: test (10000 zлата) ; PokretniEntitet: (0.0, 0.0) [100 HP] + | brzina: 1.0, ugao: 0.0

// ispis nakon pomeranja:
PokretniEntitet: (1.0, 0.0) [100 HP] + | brzina: 1.0, ugao: 0.0
Igrac: test (10000 zлата) ; PokretniEntitet: (1.0, 0.0) [100 HP] + | brzina: 1.0, ugao: 0.0
```

6. (4 poena) Napisati klasu `Igra` koja testira rad prethodno implementiranih klasa. Prvo treba napraviti igrača (ime i koordinate se unose sa standardnog ulaza) i ispisati ga:

```
Unesite ime: Playa
Unesite X koordinatu pocetka: 50.5
Unesite y koordinatu pocetka: 25.5
Igrac: Playa (10000 zлата) ; PokretniEntitet: (50.5, 25.5) [100 HP] + | brzina: 1.0, ugao: 0.0
```

Nakon toga treba napraviti niz objekata klase `DeoOpreme`, instancirati 5 objekata (vrednosti polja su date ispod u primeru izlaza) i ispisati ih na standardni izlaz po jednog u svakoj liniji sa rednim brojem na početku linije:

```
1. DeoOpreme: Viteski Mac (3000 zлата)
2. DeoOpreme: Buzdovan (1750 zлата)
3. DeoOpreme: Veliki stit (1550 zлата)
4. DeoOpreme: Drevni oklop (5000 zлата)
5. DeoOpreme: Napitak za oporavak (250 zлата)
```

Igra počinje tako što se prvo igrač priprema i kupuje sve što smatra da mu je potrebno. Najpre se ispiše ovaj meni sa svim dostupnim artiklima za kupovinu, a onda korisnik bira redni broj po izboru (ako se unese nevalidan redni broj, ponoviti postupak). Taj artikal se kupuje (poziva se `kupi()` metod klase `Igrac`). Kopirati artikal koji se kupuje prilikom prosledjivanja metodu `kupi()` - dakle meni se ne menja. Igrač kupuje sve dok ima zлата za bilo koji od ovih artikala. Na kraju se ispisuje broj instanciranih delova opreme, zatim igrač, i program se zaustavlja.

```
Imate > 10000 < zлата.
Unesite rbr artikla koji zelite da kupite: 4
Kupili ste DeoOpreme: Drevni oklop (5000 zлата)
Imate > 5000 < zлата.
Unesite rbr artikla koji zelite da kupite: 1
Kupili ste DeoOpreme: Viteski mac (3000 zлата)
Imate > 2000 < zлата.
Unesite rbr artikla koji zelite da kupite: 1
Kupili ste DeoOpreme: Buzdovan (1750 zлата)
Imate > 250 < zлата.
Unesite rbr artikla koji zelite da kupite: 5
Kupili ste DeoOpreme: Napitak za oporavak (250 zлата)
Imate > 0 < zлата.

Ukupno instancirano artikala: 10
Igrac: Playa (0 zлата) ; PokretniEntitet: (50.5, 25.5) [100 HP] + | brzina: 1.0, ugao: 0.0
    - DeoOpreme: Drevni oklop (5000 zлата)
    - DeoOpreme: Viteski mac (3000 zлата)
    - DeoOpreme: Buzdovan (1750 zлата)
    - DeoOpreme: Napitak za oporavak (250 zлата)
```

Igra pocinje! To be continued...