

R smer, Dizajn programskih jezika, grupa A, 26.1.2017.

Na *Desktop*-u napraviti direktorijum čije je ime u formatu **dpj_j1_ImePrezime_BrojIndeksa**.

Na primer, **dpj_j1_JovanPetrovic_mr14072**. Sve zadatke sačuvati u ovom direktorijumu.

NAPOMENA: 1. i 2. zadatak se rade u programskom jeziku Java, a 3. i 4. u programskom jeziku C++. Eliminacioni test primeri su obeleženi zvezdicom. Potrebno je da se strogo držite formata ispisa koji je naznačen u zadacima. Da biste osvojili poene, neophodno je da se program uspesno kompilira.

1. (20%) Korisnik prvo unosi dimenziju niza, a zatim i njegove elemente, cele brojeve, sa standardnog ulaza. Napisati program koji proverava koji su elementi niza deljivi sumom svojih cifara (za broj 0 smatrati da nije deljiv sumom svojih cifara) i ispisuje ih na standardni izlaz. Za proveru da li elementi niza zadovoljavaju pomenut uslov i njihov ispis u tom slučaju, koristiti zasebne niti. Broj niti treba da odgovara broju procesora koji su na raspolaganju.
- NAPOMENA: *Redosled ispisa ne mora biti isti kao u test primerima.*

Primer 1*	Primer 2	Primer 3	Primer 4
<pre>ULAZ: 3 100 1000 10001 IzLAZ: 100 1000</pre>	<pre>ULAZ: 4 -102 102 -1234 1234 IzLAZ: -102 102</pre>	<pre>ULAZ: 3 -10 -11 -12 IzLAZ: -10 -12</pre>	<pre>ULAZ: 3 11 -13 17 IzLAZ:</pre>

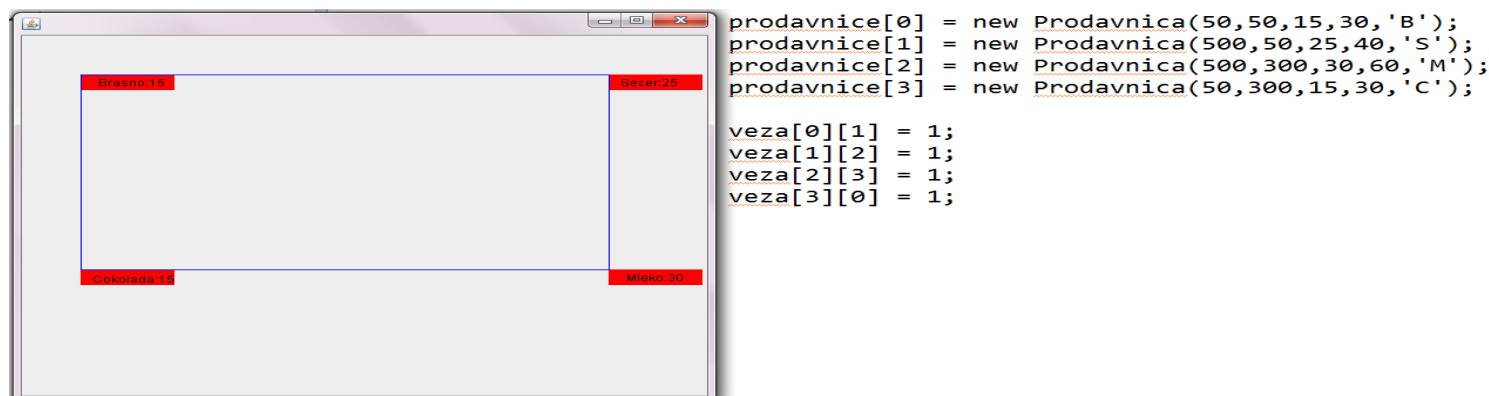
2. (30%) Vreme je spremanja slavonskih kolača. Pet domaćica pravi kolače i postoje četiri prodavnice u kojima mogu nabaviti sve potrebne sastojke: brašno, šećer, mleko i čokoladu. U svakoj mogu nabaviti tačno jedan sastojak. Napisati program koji simulira kupovinu pet domaćica. Ove domaćice kreću iz nasumično odabranih prodavnica (dozvoljeno je da više njih krene iz iste prodavnice), idu redom i kupuju potrebne sastojke, a kada sve završe kupovinu program se zaustavlja.

Prodavnice:

Prodavnice dopunjuju svoje zalihe do maksimalnog kapaciteta periodično na svakih 5 sekundi. Prodavnice su deljeni resursi, iz razloga što u datom momentu više domaćica može čekati na robu u njoj. Mehanizam čekanja u prodavnicama realizovati korišćenjem kondicione liste. Prodavnice su raspoređene na panelu dimenzije 600 x 500 kao na slici ispod, a veze između njih su zadate matricom povezanosti *veza* dimenzije 4 x 4 za koju važi da je element *veza[i][j]* jednak 1 ukoliko postoji put od prodavnice *i* do prodavnice *j*, a 0 inače. Prikazanu matricu smatrati fiksnom. Pri kreiranju prodavnica konstruktoru se prosleđuju koordinate prodavnice, trenutno stanje, maksimalni kapacitet i oznaka B, S, M ili C u zavisnosti koji sastojak se prodaje u toj prodavnici. Prodavnice grafički predstaviti kao crvene pravougaonike dimenzije 80 x 20. Na sredini pravougaonika ispisati koji se sastojak prodaje i trenutno stanje.

Domaćice:

U slučaju da prodavnica trenutno nema potrebnu količinu robe, domaćica čeka dok se zaliha ne obnovi. Kad završi kupovinu u jednoj prodavnici, domaćica kreće u sledeću prodavnicu. Domaćice se kreću u smeru kazaljke na satu. U trenutku kad nabave sve potrebne sastojke zaustavljaju se. Potrebno je kreirati pet domaćica i inicijalno ih postaviti na nasumično odabrane polazne prodavnice. Takođe, inicijalno, korpe svih domaćica su prazne, dakle, moraće da kupe odgovarajući sastojak iz polazne prodavnice. Svaku domaćicu treba implementirati kao zasebnu nit i grafički ih predstaviti kao pravougaonike dimenzije 10 x 10 koji su prazni sve dok domaćica ne nabavi potrebne sastojke. Pri kreiranju objekata konstruktoru se prosleđuju potrebne količine brašna, šećera, mleka i čokolade (random izabrani brojevi od 10 do 30).



Slika 1: Na slici je dat grafički prikaz prodavnica, a u kodu pored se mogu videti tačne koordinate, trenutno stanje, kapacitet za prodavnice i oznaka robe koja se prodaje, kao i matrica povezanosti prodavnica (prikazani su samo oni elementi koji su jednaki 1, ostali elementi su jednaki 0)

NAPOMENA: 2. zadatak možete rešiti bez pravljenja grafičkog interfejsa. U tom slučaju možete dobiti maksimalno $\frac{2}{3}$ od broja poena na ovom zadatku. U ovoj verziji rešenja, potrebno je ispisivati na standardni izlaz kada se obavi neka kupovina. Na primer:

1. domaćica je kupila potrebnu količinu brasna.
3. domaćica je kupila potrebnu količinu secera.

3. (20%) Napisati šablon klasu `Krug` u okviru prostora imena *geometrija* koja kao polje podatak ima poluprečnik kruga r , tipa T koji je parametar šablona i u njoj obezbediti:

- (a) konstruktor sa jednim argumentom
- (b) operatore poređenja $<$, $==$ (poređenje krugova se svodi na poređenje njihovih poluprečnika, pretpostaviti da za objekte tipa T postoje definisani operatori poređenja)
- (c) metode za računanje površine i obima kruga (koristiti konstantu `M_PI` definisanu u zaglavlju `cmath`), pretpostaviti da za objekte tipa T postoji definisano množenje sa konstantama i da će rezultat izračunavanja biti tipa T

Napisati program koji testira šablon klasu `Krug` za poluprečnik tipa `double`. Program prvo učitava podatke za dva kruga sa realnim poluprečnicima i ispisuje 0 ako nisu jednaki i obim i površinu za manji od unetih krugova ili 1 ako su jednaki i obim i površinu bilo kog kruga. U slučaju neispravnih ulaznih podataka ispisati -1 i prekinuti program.

Primer 1*

```
ULAZ:
4.4
5.5
IZLAZ:
0
27.646
60.8212
```

Primer 2

```
ULAZ:
4.4
4.4
IZLAZ:
1
27.646
60.8212
```

Primer 3

```
ULAZ:
2.4
2.3
IZLAZ:
1
14.4513
16.619
```

Primer 4

```
ULAZ:
4.4
-5.4
IZLAZ:
-1
```

4. (30%) Pred dan Matematičkog fakulteta potrebno je odrediti studente koji će dobiti nagradu. Napisati program koji pomaže u određivanju dobitnika na osnovu podataka o studentima (imena, prezimena i njihovih proseka) pri čemu je kriterijum izbora visina proseka.

- a) Napisati funkcional pod nazivom `Nagrada` koji za argument prihvata vrednost koja predstavlja minimalan prosek za dobijanje nagrade i služi za proveru da li neki student zadovoljava uslov za dobijanje nagrade ili ne. Ukoliko vrednost prosledena konstruktoru nije ispravan prosek, podrazumevano postaviti minimalni prosek na 9.50.
- b) Napisati šablon funkciju koja za argumente dobija kolekciju sa podacima o studentima i proizvoljan funkcional i ispisuje prezimena i imena studenata koji su dobitnici nagrade ili odgovarajuću poruku ukoliko takvih nema.
- c) Sa standardnog ulaza se unosi ukupan broj studenata, podaci o studentima i minimalni prosek za dobijanje nagrade. Podaci o studentu sadrže ime i prezime u obliku *PrezimeIme* (pretpostaviti da će ovaj format biti jedinstven) i prosek. Pretpostaviti da su svi podaci o studentima ispravno zadati. Na standardni izlaz ispisati:

1) učitane podatke o svakom studentu u obliku *Prezime Ime prosek* sortirano po ulaznom podatku *PrezimeIme*.

2) prezimena i imena svih studenata koji su dobitnici nagrade Matematičkog fakulteta.

Primer 1*

```
ULAZ:
3
PetrovicMilan 10.00
MaricAleksa 9.00
AnticNikolina 9.20
IZLAZ:
Antic Nikolina 9.2
Maric Aleksa 9
Petrovic Milan 10
ULAZ:
9.80
IZLAZ:
Petrovic Milan
```

Primer 2

```
ULAZ:
3
PetrovicMila 9.50
JokicAna 9.20
JoksicAnica 9.60
IZLAZ:
Jokic Ana 9.2
Joksic Anica 9.6
Petrovic Mila 9.5
ULAZ:
11.00
IZLAZ:
Joksic Anica
Petrovic Mila
```

Primer 3

```
ULAZ:
3
PericMilena 9.00
MaricAleksandra 9.20
LazicNikolina 9.22
IZLAZ:
Lazic Nikolina 9.22
Maric Aleksandra 9.2
Peric Milena 9
ULAZ:
9.80
IZLAZ:
Nema dobitnika nagrade.
```

Primer 4

```
ULAZ:
2
JoksicMilena 9.00
MaricAleksandra 9.20
IZLAZ:
Joksic Milena 9
Maric Aleksandra 9.2
ULAZ:
-9.80
IZLAZ:
Nema dobitnika nagrade.
```