

R smer, Dizajn programskih jezika, 25.1.2017.

Na *Desktop*-u napraviti direktorijum čije je ime u formatu `dpj_jan1_ImePrezime_BrojIndeksa`.

Na primer, `dpj_jan1_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 uspešno kompilira.

1. (20%) Napraviti aplikaciju koja paralelno generiše random metodom n pozitivnih četvorocifrenih brojeva i određuje među njima broj koji ima najveću apsolutnu vrednost razlike cifre jedinica i hiljada. Za generisanje određenog broja i određivanje apsolutne vrednosti razlike njegove cifre jedinica i hiljada koristiti zasebne niti. Broj niti treba da odgovara broju procesora koji su na raspolaganju. Nakon obrade brojeva, glavna (`main`) nit treba da prikaže traženi broj. Ukoliko ima više takvih, prikazati od njih onaj za koji je `random_seed + i` najmanje. Pretpostaviti da su ulazni podaci ispravni.

GUI: Za osnovni kontejner aplikacije upotrebiti Border Pane dimenzije 200×200 sa marginama veličine 20. U levu oblast postaviti tekstualno polje za unos broja n , u centralnu tekstualno polje za unos broja koji predstavlja `random_seed`, u desnu dugme za učitavanje, a u donju labelu u kojoj treba prikazati tekst sa rezultatom (pogledati sliku). Korisnik unosi u levo tekstualno polje broj n , u centralno veličinu za `random_seed`, nakon čega pritiskom na dugme započinje generisanje brojeva i obrada. Rezultat obrade prikazati u labeli.

UPUTSTVO: Izvlačenje podataka iz tekstualnih polja i određivanje rezultata implementirati kao metod za akciju nad dugmetom. Za generisanje i -tog broja, koristiti objekat klase `Random` sa argumentom `random_seed + i`. Četvorocifreni brojevi se mogu dobiti od random generisanog broja iz intervala $[0, 9000)$ kome je dodat broj 1000.



Slika 1: Na slici je dat izgled aplikacije i primer za vrednosti 5 za n i 10 za `random_seed` pri čemu se generišu brojevi 5113 1738 2866 8492 9615.

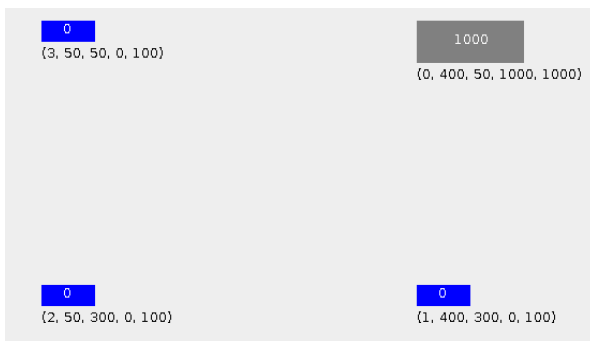
NAPOMENA: Ovaj zadatak možete rešiti bez pravljenja grafičkog interfejsa. U tom slučaju možete dobiti maksimalno $\frac{3}{5}$ od broja poena na ovom zadatku. U ovoj verziji rešenja, potrebno je sa standardnog ulaza učitati vrednosti za n i `random_seed` i rezultat ispisati na standardni izlaz.

2. (30%) Napisati program koji simulira centralizovan rad pošte. Poštari dobijaju zaduženja u centralnoj pošti u vezi sa pošiljkama, a kada raznesu sve pošiljke u pomoćne pošte, program se zaustavlja.

Pošte: Postoji jedna centralna, velika pošta i tri pomoćne, male pošte. Poštari donose pošiljke iz centralne pošte u pomoćne pošte, odakle se na svakih 5 sekundi šalje najviše 10 pošiljki na adrese primaoca. Pošte su deljeni resursi, iz razloga što u datom momentu više poštara može doneti pošiljke (u slučaju pomoćnih pošti), odnosno uzeti pošiljke (u slučaju centralne pošte). Mehanizam čekanja u poštama realizovati korišćenjem kondicione liste. Poštu karakteriše njen redni broj (0 za centralnu poštu i 1, 2 ili 3 za pomoćne poste), njene koordinate koje određuju gde će biti prikazana, trenutno stanje (ukupan broj pošiljki koje u datom trenutku treba da se dostave primaocima) i kapacitet (maksimalan broj pošiljki u pošti u bilo kom trenutku). Centralna pošta ima stanje i kapacitet jednak 1000 što predstavlja ukupan broj pošiljki koje treba da se dostave u pomoćne pošte. Pomoćne pošte imaju pri kreiranju stanje 0 i kapacitet 100.

Poštari: Poštari kreću iz centralne pošte odakle prenose nasumično odabran ceo broj pošiljki iz segmenta $[1, \min\{20, \text{stanje}\}]$ u nasumično odabranu pomoćnu poštu. Kada stignu u pomoćnu poštu, isporučuju pošiljke i vraćaju se u centralnu poštu po novu turu pošiljki. U slučaju da u pomoćnoj pošti nema prostora za prijem pošiljki, čekaju dok se ne oslobodi prostor. U trenutku kada ne bude više bilo pošiljki u centralnoj pošti, zaustavljaju se. Potrebno je kreirati 7 poštara i svakog poštara treba implementirati kao zasebnu nit.

GUI: Pošte su raspoređene na panelu dimenzije 600×500 (slika na sledećoj strani). Centralnu poštu grafički predstaviti kao sivi pravougaonik dimenzije 100×40 , a pomoćne kao plave pravougaonike dimenzije 50×20 . Na sredini pravougaonika ispisati trenutno stanje pošiljki. Inicijalno poštare postaviti na centralnu poštu i grafički ih predstaviti kao krugove prečnika 5 koji su puni kada prenose pošiljke iz centralne pošte, a prazni kada se vraćaju u centralnu poštu.



Slika 2: Na slici je dat grafički prikaz poštanskog sistema. U pravougaonicima je prikazano inicijalno stanje broja pošiljki, a ispod njih redni broj pošte, tačne koordinate, stanje, i kapacitet.

NAPOMENA: Ovaj 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 određeni poštar isporuči pošiljke pomoćnoj pošti. Na primer: Poštar 5 je predao pošiljke pomoćnoj pošti 1.

3. (20%) Napisati šablonsku klasu *BMI* u okviru prostora imena *indeks* koja kao polja podatke ima visinu u metrima i masu u kilogramima. Oba podatka su tipa *T* koji je parametar šablona. U klasi obezbediti:

- konstruktor sa dva argumenta
- metod za računanje indeksa telesne mase (*BMI*) koji se dobija deljenjem mase kvadratom visine (pretpostaviti da za objekte tipa *T* postoje definisane operacije množenja i deljenja i da će rezultat izračunavanja biti tipa *T*)
- operatore poređenja *<*, *==* (poređenje objekata se svodi na poređenje indeksa telesne mase (pretpostaviti da za objekte tipa *T* postoje definisani operatori poređenja)

Program za dva učitana objekta klase *BMI<double>* proverava da li su jednaki i ukoliko jesu štampa indeks telesne mase prvog objekta, a ukoliko nisu ispisuje apsolutnu vrednost razlike njihovih indeksa telesne mase. Pretpostaviti da su ulazni podaci ispravni, a da se prilikom učitavanja objekata najpre učitava visina pa masa.

Primer 1*	Primer 2	Primer 3	Primer 4
<pre> ULAZ: 1.8 70 1.7 65 IZLAZ: 0.366398 </pre>	<pre> ULAZ: 1.72 101.5 1.8 89 IZLAZ: 6.83995 </pre>	<pre> ULAZ: 1.8 81 1.5 56.25 IZLAZ: 25 </pre>	<pre> ULAZ: 1.8 72 1.8 72 IZLAZ: 22.2222 </pre>

4. (30%) Marija je krenula u kupovinu maturalne haljine. Napisati program koji joj pomaže u kupovini na osnovu podataka o raspoloživim modelima u prodavnici, pri čemu je kriterijum izbora tražena veličina i cena.

- Napisati klasu pod nazivom *Haljina* koja sadrži podatke o modelu haljine u prodavnici. Klasa kao polja podatke ima naziv modela (string), veličinu (ceo broj) i cenu (realan broj).
- Napisati funkcional pod nazivom *Odgovarajuca* koji za argumente prihvata vrednost koja predstavlja traženu veličinu i vrednost koja predstavlja maksimalnu cenu i služi za proveru da li neka haljina zadovoljava uslove kupovine ili ne. Ukoliko vrednost argumenta za veličinu nije ispravna, podrazumevano postaviti da je tražena veličina 36, u slučaju da vrednost za cenu nije ispravna, podrazumevano postaviti da je tražena cena 2000.
- Napisati šablon funkciju koja za argumente dobija kolekciju objekata klase *Haljina* i proizvoljan funkcional i ispisuje modele haljina koje su tražene veličine i u zadatom cenovnom rangi ili odgovarajuću poruku ukoliko takvih haljina nema.
- Sa standardnog ulaza se unosi ukupan broj modela haljina, podaci o njima i tražena veličina i maksimalna cena. Podaci o haljinama se unose u sledećem redosledu *naziv modela*, *veličina*, *cena*. Pretpostaviti da su svi podaci o haljinama ispravno zadati i da će naziv modela uvek biti jedna reč. Na standardni izlaz ispisati modele haljina koje su odgovarajuće veličine i u zadatom cenovnom rangi.

Primer 1*	Primer 2	Primer 3	Primer 4
<pre> ULAZ: 4 PS 34 3999 Koton 36 1999 Nike 40 4500 PS 42 2500 36 2500 IZLAZ: Koton </pre>	<pre> ULAZ: 3 C&A 34 1599.99 Nike 40 4500 NY 40 3500 40 4500 IZLAZ: Nike NY </pre>	<pre> ULAZ: 2 C&A 38 2599.99 PS 36 1500 -26 4500 IZLAZ: PS </pre>	<pre> ULAZ: 4 C&A 38 3000 PS 34 1599.99 NY 40 3500 PS 36 1500 38 -1000 IZLAZ: Nema traženih modela. </pre>