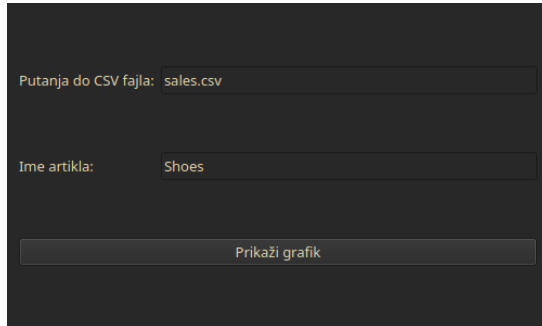
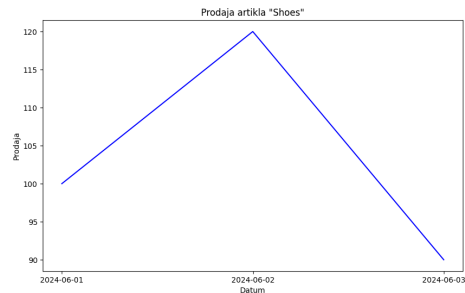


R smer, Dizajn Programskih Jezika, Jun 2024

1. **[Python]** Napisati program kojim se iscrtava grafik prodaje artikala koji zadaje korisnik. Grafik treba da bude iscrtan plavom bojom. Napraviti odgovarajući grafički interfejs na kom postoje dva polja za unos: putanje do csv fajla u kojem se nalaze sve prodaje i artikla za koji se iscrtava grafik, i jedno dugme (pogledati sliku 1, zanemariti tamnu pozadinu). Klikom na dugme započinje iscrtavanje grafika i prikaz istog (pogledati sliku 2). Pretpostaviti da će zadati podaci biti ispravni.



Slika 1: Interfejs sa poljima za unos putanje do CSV fajla i imena artikla



Slika 2: Prikaz grafika prodaje nakon klika na dugme "Prikaži grafik"

2. **[Go]** Implementirati konkurentan program koji obradom n-dimenzionalnog vektora transformiše svaku reč unetu sa standardnog ulaza. Korisnik unosi broj reči, a zatim te reči. Svaka uneta reč se obradi na način da se dupliraju svi samoglasnici unutar nje. Program treba da koristi zasebne gorutine za obradu svake reči, pri čemu broj gorutina treba da odgovara broju dostupnih procesora. Nakon obrade svih reči, program treba da ispiše transformisane reči na standardni izlaz.

Napomena: *Pretpostaviti da su svi unosi ispravni i da reči sadrže samo slova bez dijakritičkih znakova.*

Primer 1	Primer 2	Primer 3	Primer 4
Ulaz: 3 jabuka banana kruska Izlaz: jaabuukaa baanaanaa kruuskaa	Ulaz: 3 Leptir Cvet Okean Izlaz: Leeptiir Cveet O0keean	Ulaz: 2 krst prst Izlaz: krst prst	Ulaz: 5 a aa b bb aao Izlaz: aa aaaa b bb aaaaoo

3. **[Go]** Letnje vreme je idealno za spremanje poslastica. Pet prijateljica je odlučilo da organizuje okupljanje i tom prilikom će napraviti različite poslastice. Međutim, svakoj od njih nedostaje nešto od sledećih sastojaka: jagode, sladoled, čokolada ili banane. Postoje četiri prodavnice u kojima mogu nabaviti te sastojke. U svakoj mogu nabaviti tačno jedan sastojak. Napisati program koji simulira kupovinu ovih pet prijateljica. One 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 (500 grama) periodično na svakih 5 sekundi. Prodavnice su deljeni resursi, iz razloga što u datom momentu više kupaca može čekati na robu. Mehanizam čekanja u prodavnicama realizovati korišćenjem uslovnog reda čekanja. Svaka prodavnica od informacija ima jedinstveni identifikacioni broj, trenutno stanje, maksimalni kapacitet i oznaku J, S, C ili B u zavisnosti koji sastojak se prodaje u toj prodavnici.

Prijateljice: U slučaju da prodavnica trenutno nema potrebnu količinu robe, prijateljica čeka dok se zaliha ne obnovi. Kad završi kupovinu u jednoj prodavnici, kreće u sledeću nasumično odabranu prodavnicu. Potrebno je kreirati pet prijateljica i inicijalno ih postaviti na nasumično odabrane polazne prodavnice. Takođe, inicijalno, korpe svih njih su prazne. Svaku prijateljicu treba implementirati kao zasebnu gorutinu. Pri kreiranju prijateljice proslediti potrebne količine jagoda, sladoleda, čokolada ili banana (nasumično izabrani brojevi od 0 do 1000 koji se odnose na gramažu odgovarajućeg proizvoda).

Modelovati opisan sistem kupovine i prikazivati informacije kad god neka od prijateljica uđe u prodavnicu, kad se obavi kupovina i finalno, kad neka od prijateljica završi sa nabavkom svih sastojaka.

4. [C++] Napisati šablonsku klasu *Polinom* koja kao polje ima niz koeficijenata polinoma (pozicija i predstavlja koeficijent ispred x^i). Koeficijenti i svi ostali podaci su tipa T koji je parametar šablona. U klasi obezbediti:

- Konstruktor koji prima niz koeficijenata (ako se pozove bez argumenata, podrazumeva se niz sa jednim elementom čija je vrednost 1).
- Metod **calculate** koji kao argument prima x i vraća izračunatu vrednost polinoma za to x .

Napisati program koji učitava *Polinom*<float>, tj. broj koeficijenata i njihove vrednosti, i x . Na standardi izlaz se ispisuje vrednost tog polinoma. Pretpostaviti da će ulazni podaci biti ispravni.

Primer 1

```
Ulaz:
3
1.5 -2.0 0.5
2.0
Izlaz:
-0.5
```

Primer 2

```
Ulaz:
4
-1.0 0.0 2.5 -3.5
-1.5
Izlaz:
16.4375
```

Primer 3

```
Ulaz:
1
3
0
Izlaz:
3.0
```

Primer 4

```
Ulaz:
2
-0.5 1.0
1.0
Izlaz:
0.5
```

5. [C++] Napisati program koji pomaže u kupovini bioskopske karte, pri čemu je kriterijum, pored cene, ocena filma.

- Napisati klasu *Film* koja sadrži podatke o filmovima koji se emituju u bioskopu: naziv (string), cena (neoznačen ceo broj) i ocena (realni broj dvostruke tačnosti u dinarima).
- Napisati klasu koja predstavlja funkcijski objekat *BestFilm* koji pri konstrukciji pamti količinu novca koju imamo na raspolaganju i trenutnu najbolju nađenu ocenu filma postavlja na 0. Pri funkcijskom pozivu ovog objekta se prosleđuje instanca klase *Film* i proverava se da li je taj film najbolji do sada viđen film koji možemo kupiti. Ukoliko jeste, ažurira se vrednost funkcijskog objekta za trenutnu najbolju nađenu ocenu filma.
- Napisati šablon funkciju koja za argumente dobija proizvoljnu kolekciju objekata klase *Film* i količinu novca koju imamo na raspolaganju i ispisuje naziv najbolje ocenjenog filma koji možemo da kupimo za taj novac, ili poruku ukoliko takvog filma nema u bioskopu.
- Sa standardnog ulaza se učitava ukupan broj filmova, podaci o njima kao i budžet za kupovinu karte. Podaci o jednom filmu se unose u redosledu *naziv*, pa *ocena* pa *cena karte*. Pretpostaviti da su svi podaci o igračkama ispravno zadati i da će naziv filma uvek biti jedna reč. Na standardni izlaz ispisati naziv najbolje ocenjenog filma koji možemo da kupimo zadatim budžetom, ili poruku ukoliko takvog filma nema u bioskopu.

Primer 1

```
Ulaz:
3
Gladiator 8.5 100
Matrix 8.7 200
Inception 8.8 300
250
Izlaz:
Matrix
```

Primer 2

```
Ulaz:
3
Inception 8.8 500
Avatar 8.5 450
Titanic 9.5 300
400
Izlaz:
Titanic
```

Primer 3

```
Ulaz:
4
Inception 8.8 500
Avatar 8.5 450
Titanic 9.5 300
Matrix 8.7 350
250
Izlaz:
Nema adekvatan film!
```

Primer 4

```
Ulaz:
4
Frozen 8.2 200
Moana 7.8 150
ToyStory 9.1 350
LionKing 8.9 300
300
Izlaz:
LionKing
```