

1. Omogućiti korisniku da unosi reči sve dok ne unese reč koja se završava slovom a. Potom ispisati na konzolu ukupan broj pogrešnih unosa.

Primer izvršavanja:

>> ispit

>> programiranje

>> ocena

>> Broj pogrešnih unosa: 2

2. Napiši program koji od korisnika traži da unosi brojeve sve dok ne unese neparni broj. Kada korisnik unese neparni broj, izvršavanje programa se prekida i korisniku se ispisuje suma prethodno unetih (parnih) brojeva.

3. Napraviti fajl sa nazivom radnici.txt koji u svakom redu sadrži ime radnika, pol i platu. Potom na osnovu fajla izračunati prosečnu platu svih zaposlenih osoba ženskog pola. Rezultat ispisati na konzolu.

Primer fajla:

Pera|M|35000

Jova|M|63000

Ana|Z|45000

Maja|Z|50000

Rezultat: 47500

4. Napiši program koji kombinuje sadržaj dva fajla i snima ga u treći fajl. U prvom fajlu se nalaze korisnička imena i lozinke prodavaca, a u drugom fajlu se nalaze nizovi cena artikala koje su prodavci prodali. Pri tome prvom redu u fajlu sa korisničkim imenima i lozinkama odgovara prvi red u fajlu sa računima, drugom redu u fajlu sa korisničkim imenima i lozinkama odgovara drugi red u fajlu sa računima itd. Svaki red trećeg fajla treba da sadrži korisničko ime prodavca, ukupnu cenu robe koju je prodao i prosečnu cenu artikla koji je prodao.

Primer izvršavanja programa:

Ako su dati fajlovi

korisnici.txt

pera|peric

jova|jovic

steva|stevic

i

racuni.txt

100|200|150|150|300|100

50|100|100|50

300|400|200|100|400

Formirani fajl treba da bude:

statistika.txt

pera|1000.0|166.66666666666666

jova|300.0|75.0

steva|1400.0|280.0

Dodatak. Izmeni formatiranje tako da se prikazuju samo prve dve decimale:

statistika.txt

pera|1000.00|166.67

jova|300.00|75.00

steva|1400.00|280.00

5. Napiši program za evidenciju studenata. Za svakog studenta unosi se: ime, prezime, broj indeksa i godina studija. Podaci za svakog studenta se čuvaju u jednom redu u fajlu, u formatu ime|prezime|indeks|godina_studija. Ime i prezime moraju da počinju velikim slovom. Indeks mora da bude jedinstven (ako se ponovi ista vrednost za indeks, omogućiti korisniku da ponovo unese).

Primer nakon 2 unosa sa tastature:

Ime: Mihajlo

Prezime: Pupin

Broj indeksa: AA15

Godina upisa: 1885

Ime: ervin

Prezime: sredinger

Broj indeksa: BB18

Godina upisa: 1906

Studenti.txt

Mihajlo | Pupin | AA15 | 1885

Ervin | Sredinger | BB18 | 1906

6. Omogućiti korisniku da unosi imena u listu. Imena se unose sve dok korisnik ne unese 'q'. Nakon prekida unosa imena sortirati ih po abecednom redu, potom pozvati funkciju koja izbacuje iz liste imena koja počinju na slovo A ili se završavaju na slovo A/a.

7. Omogućiti korisniku da unosi brojeve, ukoliko je broj deljiv sa 4 dodati ga u sumu. Kada suma pređe 1000 program se završava i ispisuje korisniku rezultat.

8. Napiši program za registrovanje novih proizvoda u prodavnici. Prodavac može da doda novi proizvod, pri čemu unosi naziv, cenu i raspoloživu količinu proizvoda. Nakon što je dodat novi proizvod ispisuje se spisak svih proizvoda u prodavnici. Dodavanje proizvoda se ponavlja dok prodavac ne unese „quit“. Ukoliko prodavac unese „quit“ bilo za naziv, cenu ili količinu proizvoda, prekida se izvršavanje programa. Nazivi, cene i količine proizvoda čuvaju se u fajlu proizvodi.txt.

9. Stanje na računu. Iz log fajla (bank_log.txt) čitati podatke o korisnicima. Podaci su dati u obliku: ime_korisnika tip_akcije iznos.

Tipovi akcija su sledeći:

income - priliv sredstava na korisnikov račun

withdrawal - korisnik je podigao novac sa računa

-> Za sve korisnike odrediti stanje na računu.

-> Kreirati izveštaj stanja na računu za unetog korisnika. Izveštaj ispisati na konzoli u proizvoljnom formatu.

10. Dodatak na zadatak 9. Ispisati korisnike u novi fajl sa trenutnim stanjem na računu.

Napomena: svaki zadatak je potrebno implementirati kroz funkcije.
