

## ZADACI ZA NTP

1. Napisati program u kome se čitaju podaci o proizvodima u jednoj prodavnici auto-djelova iz postojeće datoteke **proizvodi.txt** i kreira niz struktura tipa **proizvod** ( naziv, proizvođač, cijena, količina),a zatim štampa:
  - ukupan broj proizvoda u datoteci,
  - ukupna vrijednost svih proizvoda u prodavnici,
  - spisak proizvoda čija je količina 0,
  - broj proizvoda koje proizvodi FIAT,
  - spisak proizvoda sortiranih po cjeni.
2. Napisati program u kome se čitaju podaci o apartmana jedne turističke agencije iz postojeće datoteke **apartmani.txt** i kreira niz struktura tipa **apartman** ( naziv, mjesto, broj kreveta, cijena), a zatim štampa:
  - ukupan broj apartmana u datoteci,
  - prosječna cijena apartmana u toj agenciji,
  - broj apartmana u mjestu Paralia,
  - naziv apartmana za 2 osobe sa najmanjom cjenom,
  - spisak apartmana sortiranih po cjeni.
3. Napisati program u kome se čitaju podaci o radnicima iz postojeće datoteke **radnici.txt** i kreira niz struktura tipa **radnik** (ime, prezime, broj godina radnog staža, stepen stručne spreme i plata), a zatim štampa:
  - ukupan broj radnika u datoteci,
  - prosječna plata radnika,
  - broj radnika koji imaju više od 5 godina radnog staža,
  - imena radnika sa najvišim stepenom stručne spreme,
  - spisak radnika sortiranih po imenima.
4. Napisati program u kome se čitaju podaci o učenicima iz postojeće datoteke **učenici.txt** i kreira niz struktura tipa **učenik** (ime, prezime, razred (1-4), uspjeh, prosječna ocjena), a zatim štampa:
  - ukupan broj učenika u datoteci,
  - broj učenika 4. razreda,
  - imena učenika koji imaju odličan uspjeh,
  - ime učenika koji ima najbolju prosječnu ocjenu,
  - imena učenika sortiranih po razredima.
5. Napisati program u kome se čitaju podaci o klijentima iz postojeće datoteke **klijenti.txt** i kreira niz struktura tipa **klijent** (ime, prezime, mjesto, telefonski broj i posljednji račun), a zatim štampa:
  - ukupan broj klijenata,
  - prosječan račun za posljednji mjesec,
  - broj klijenata iz Zenice,
  - ime i broj klijenata sa najvećim računom,
  - spisak klijenata sortiranih po prezimenu.
6. Kreiranje tekstualne datoteke „**Podaci.txt**“ u radnom direktorijumu i upis jednog reda koji sadrži: ime, pol, godište i prosjek. Omogućiti dodavanje novih podataka u datoteku. Odrediti broj ženskih osoba i osoba sa prosjekom većim od 4.50 u datoteci **Podaci.txt**.
7. U datotekama **grupa1.txt**, **grupa2.txt** i **grupa3.txt** prati se rad učenika na vježbama iz programiranja. U svakom redu datoteka je ime učenika, prosječna ocjena vježbi i broj vježbi koje je učenik propustio. Pročitati sve podatke i u datoteku **izvjestaj.txt** upisati sortirane učenike, njihove zaključne ocjene i komentar da li je učenik zadovoljio minimum rada na vježbama (manje od 5 propuštenih vježbi).

8. U datoteci **ulaz.txt** nalazi se broj elemenata niza. Pročitati taj broj, alocirati niz te dužine i unjeti vrijednosti elemenata tog niza. Izračunati prosječnu vrijednost, proširiti niz za jedno mjesto i ubaciti prosjek u niz. U novu datoteku **Izlaz.txt** upisati sortirani niz.
9. Neopravdani izostanci učenika čuvaju se u datoteci **Izostanci.txt**. Pročitati ih i napraviti niz. Pomoću funkcija izračunati: ukupan broj izostanaka u odjeljenju, prosječan broj izostanaka, koliki je najveći broj i koliko ima učenika bez neopravdanih izostanaka.
10. Napisati funkciju koja računa prosječnu vrednost niza cjelih brojeva i broj elemenata jednakih 5. Glavna funkcija koristi datoteku **Ocjene.txt** u kojoj se čuva broj učenika u odeljenju (u prvom redu) i u svakom sljedećem: ime učenika, prezime, strani jezik koji uči (engleski/njemački) i ocjena. Štampati prosjecne ocjene iz engleskog i njemačkog i broj petica iz oba jezika.

11. Data je datoteka **ispiti.txt** čiji redovi imaju sljedeću strukturu:

**brindexa      predmet      ocjena**

Kreirajte ovu datoteku koristeći Notepad i napunite je nekim testnim podacima. **predmeti** su zadani imenom predmeta.

Zatim napravite program koji na ulazu traži broj predmeta, te ispisuje prosječnu ocjenu i prolaznost (procenat studenata koji su položili tj. dobili ocjenu 6 ili više). Također treba program ispisat ukupan broj predmeta koje je student odslušao, broj položenih, te prosječnu ocjenu.

12. Data je tekstualna datoteka **pismo.txt**. Napisati funkcije za:

- a) ispis datoteke na standardni izlaz zadržavajući strukturu po redovima;
- b) prepis datoteke u novu mijenjajući mala i velika slova;
- c) ispisati datoteku u obrnutom redosljedu na izlaz (prva postaje zadnja, druga postaje pretposljednja riječ...)
- d) ispitati da li je neka riječ palindrom (isto značenje kada se čita i u obrnutom smjeru)

13. Zadatak zahtijeva pisanje programa koji će dešifrovati poruke koje su šifrirane kako je opisano u narednom primjeru. Dat je šifrirani tekst: saoreecessinntfi. Potrebno je primjetiti da se sastoji od šesnaest slova. Tekst je moguće postaviti u matricu veličine 4×4 čime se dobije:

```
s a o r
e e c e
s s i n
n t f i
```

Tekst će se dešifrovati ukoliko se znakovi poredaju na osnovu kolona odozgo prema dolje, tj. prvo znakovi iz prve kolone, potom znakovi iz druge kolone itd. Na osnovu navedenog, tražena riječ u primjeru je: sesnaestocifreni. Program otvara fajl pod nazivom **"test.txt"**. U fajlu se nalaze šifrirani tekstovi jedan ispod drugog. Program treba ispisati dešifrovane tekstove svaki u zasebnoj liniji. Ukoliko broj znakova ulaznog teksta nije kvadrat nekog broja program ispisuje riječ GRESKA.

**test.txt**

1. saoreecessinntfi
2. pmrrj ie2
3. dobar tekst
4. borj
5. Bjsr aoOm

Output:

sesnaestocifreni  
primjer 2  
GRESKA  
broj  
Broj Osam

#### 14. Problem

U zemlji BaNaNaZeMlJA, u gradu BaNaNaGrAd radio kao gradonačelnik veseli Zenit. Iako je Zenit bio jako savjestan čovjek i zabavu nije miješao s poslom, tu i tamo dogodilo bi mu se da se previše opusti u društvu i da koju popije. Sve i ne bi bilo tako grozno, da Zenit nije ostao bez službenog vozača zbog tzv. «globalne štednje», pa se je eto morao sam snalaziti do kuće. A policija u tome gradu, kao i u čitavoj državi, jedva čeka da zaustavi i kazni nekoga, a pogotovo one «veselije». Pa eto tako, došlo vrijeme da se kući krene, a Zenit «veseliji» od drugih, a doma se mora jer žena u zadnje vrijeme baš i nema razumijevanja za ove «teške dane». Sva sreća da je Zenit «velika faca» pa ima malo i veza naokolo

te lako sazna gdje je prometna policija. E da mu je samo još brz dolazak kući, sve bi se riješilo, zar ne!? Zadatak je naći najkraći put (najmanje prijeđenih polja) od početnog mjesta gdje se nalazi Zenit (M), do njegove kuće (K) tako da se gradom krećemo samo «povezanim» cestom (.), a da pritom nemamo «u blizini» prometnu policiju (P).

### Napomene

- Dvije ceste su «povezane» (tj. polja su susjedna) ako imaju zajedničku stranicu.
- «U blizini» znači da cesta kojom prolazimo **ne smije** imati susjedno polje na kome se nalazi prometna policija.
- Ne mora postojati jedinstveno minimalno rješenje puta do kuće.
- Iz ulaznih podataka uvijek će se moći dobiti rješenje.

### Ulazni podaci

U prvom retku tekstualne datoteke Zenit.in nalaze se dva broja odvojena jednim razmakom, M i N iz opisa zadatka (M je broj redaka, a N broj stupaca pravouglo mreže grada), za koje vrijedi sljedeće:  $1 \leq M, N \leq 100$ .

U sljedećih M redova nalazi se u svakom od njih po N znakova, također prethodno opisani.

### Izlazni podaci

U prvom i jedinom retku tekstualne datoteke Zenit.out potrebno je ispisati **minimalni** broj polja do kuće (broj polja koja predstavljaju cestu).

### Primjeri

Zenit.in	Zenit.in
4 4	5 8
xPxP	PxxxPxxx
M.x.	.x.....K
....	M..xxx.x
x..K	.x.....x
	P.x.xxxx
Mile.out	Mile.out
4	9

15. Odrediti broj minimalnih pokrivajućih stabala datog povezanog težinskog grafa u kome se ni jedna težina grane ne ponavlja više od 4 puta.

### Ulaz:

(Ulazni podaci se nalaze u datoteci **stabla.in**) U prvom redu ulaza su zapisani brojevi  $N$  i  $M$  ( $1 \leq N, M \leq 5 \times 10^4$ ), broj čvorova i broj grana datog grafa. U svakom od narednih  $M$  redova zapisana su tri cjela broja  $A$ ,  $B$  i  $W$  ( $1 \leq A, B \leq N$ ,  $1 \leq W \leq 2^{30}$ ), koji označavaju da između čvorova  $A$  i  $B$  postoji grana težine  $W$ .

### Izlaz:

(Izlazne podatke upisati u datoteku **stabla.out**) U izlaznu datoteku ispisati ostatak traženog broja minimalnih pokrivajućih stabala datog grafa koji se dobija pri djeljenju sa 1000003.

### Primjer 1:

stabla.in	stabla.out
3 4	5
1 2 6	
1 2 6	
2 3 6	
3 1 6	
3 3 8	

### Objašnjenje.

Sve grane su iste težine, između čvorova 1 i 2 postoje dvije grane, a postoji i grana kojoj su oba kraja u čvoru 3.

### Primjer 2:

stabla.in	stabla.out
6 8	
1 2 1	
2 3 2	
3 4 3	
4 5 3	
5 6 2	
6 1 1	
2 6 1	
3 5 3	