

## ZADACI ZA VJEŽBU IZ OSNOVA RAČUNARSTVA II

1. Kreirati algoritam i pseudokod za izračunavanje vrijednosti  $Y=X^N$ , pri čemu su  $X$  i  $N$  ulazni podaci.  $X$  je realan, a  $N$  cio broj. Prilagoditi algoritam da radi i za negativno  $N$ . Potrebno je napraviti algoritam za stepenovanje broja  $X$  brojem  $N$  koji će se implementirati na hardverskoj platformi (na primjer mikrokontroleru) koji podržava samo četiri osnovne operacije (sabiranje, množenje, oduzimanje i dijeljenje).
2. Nacrtati algoritamsku shemu i napisati pseudokod koji za ulazni podatak ima prirodni broj  $N$  i koji vrši sumiranje prirodnih brojeva, počev od  $1$  pa naviše, sve dok je ta suma manja od  $N$ . Štampati dobijenu sumu, kao i broj sumiranih članova.
3. Nacrtati algoritamsku shemu i napisati pseudokod koji za ulazni podatak ima prirodni broj  $N$  i koji određuje i štampa koliko puta se u broju  $N$  pojavljuje cifra  $7$ . Na primjer, ukoliko se unese  $N=1772$  onda je broj pojavljivanja cifre  $7$  jednak  $2$ .
4. Nacrtati algoritamsku shemu i napisati pseudokod koji računa sumu svih prirodnih brojeva manjih od  $1000$  čiji je zbir cifara jednak  $6$ . Na izlazu štampati tu sumu.
5. Kreirati algoritam i napisati pseudokod kojim su učitavaju dva prirodna broja  $m$  i  $n$ , i kojim se određuje najveći zajednički delilac (NZD) ova dva broja uz pomoć Euklidovog algoritma, koji se može opisati sa dva koraka:
  - (1) Ako je  $m=n$  tada je  $NZD=m$  i to je kraj algoritma;
  - (2) Veći od ta dva broja postaje jednak razlici većeg i manjeg i vraćamo se na korak 1.
6. Nacrtati algoritamsku shemu i napisati pseudokod koji izračunava približnu vrijednost broja  $\pi$  koristeći sljedeću sumu:

$$\pi = 4 - \frac{4}{3} + \frac{4}{5} - \frac{4}{7} + \frac{4}{9} - \frac{4}{11} + L = \sum_{n=0}^{+\infty} \frac{4(-1)^n}{2n+1}$$

Sumiranje prekinuti kada apsolutna vrijednost opšteg člana sume postane manja od  $10^{-4}$ .

7. Nacrtati algoritamsku šemu kojom se unosi matrica  $A$ , dimenzija  $M \times M$ , i koja od date matrice formira vektor  $B$  nadovezujući kolone matrice  $A$ , počev od prve pa do poslednje. Na izlazu štampati dobijeni vektor.
8. Kreirati algoritam za određivanje da li je niz cijelih brojeva  $X$ , čija dužina i elementi predstavljaju ulazne podatke algoritma, sortiran u rastući poredak. Niz je sortiran u rastući poredak ukoliko je svaki element niza, osim prvog, veći od prethodnog elementa. Primjer sortiranog niza je:  $4\ 7\ 11\ 21\ 29\ 35\ 57$ .
9. Data su dva niza cijelih brojeva:  $X$  od  $N$  elemenata i  $Y$  od  $M$  elemenata. Elementi nizova su sortirani u rastući poredak, pri čemu nema ponavljanja elemenata u tim nizovima. Sastaviti algoritam kojim se ova dva niza sastavljaju u treći niz  $Z$ , koji takođe treba da bude sortiran u rastući poredak. U nizu  $Z$  takođe ne smije biti ponavljanja elemenata. *Primjer:* Neka je korisnik je unio nizove  $X=2, 5, 8, 9, 14, 23$  i  $Y=-6, -1, 2, 4, 14, 19$ , onda bi niz  $Z$  trebao biti  $Z=-6, -1, 2, 4, 5, 8, 9, 14, 19, 23$ .
10. Nacrtati algoritam kojim se učitava niz cijelih brojeva, sa najviše 30 članova, i koji vrši obrtanje redosljeda elemenata vektora, tj. zamjenjuje mjesta prvom i posljednjem članu, drugom i pretposljednem itd.. Program napisati tako da vrši zamjenu elemenata datog niza bez korišćenja pomoćnih nizova. *Primjer:* Ukoliko smo unijeli niz  $X=1,32,41,6,71,13$ , onda bi nakon izvršenja programa redosljed trebao biti  $X=13,71,6,41,32,1$ .
11. Kreirati algoritam koji nakon unosa broja  $M$  kao i 20 brojeva, ispisati koliko ima unijetih brojeva koji su manji ili jednaki sa  $M$ . Prilikom unosa provjeravati svaki broj.
12. Nacrtati algoritam kojim se unosi niz  $X$  od  $N$  prirodnih brojeva. Potrebno je formirati novi niz  $Y$ , iste dužine, koji na poziciji  $Y[i]$  ima vrijednost sume elemenata niza  $X$  od početka niza do elementa  $X[i]$ . Odštampati niz  $Y$ . Nacrtajte algoritam ili napišite pseudo-kod za rješavanja zadatog problema.

13. Nacrtati algoritam koji traži sumu svih prirodnih brojeva manjih od 20000 koji su potpuni kubovi, tj. jednaki su kubu nekog prirodnog broja. Napomena: Brojevi 1,8,27,64...(1<sup>3</sup>, 2<sup>3</sup>, 3<sup>3</sup>, 4<sup>3</sup>,...) su potpuni kubovi.
14. Kreirati algoritam i napisati pseudokod koji učitava string **S** i koji provjerava da li taj string može predstavljati binarni zapis nekog broja. Štampati odgovarajuću poruku.
15. Kreirati algoritam i napisati pseudokod koji učitava cijeli broj **N** i koji konvertuje taj broj u string **S**. Recimo, ukoliko se učitava broj **N=-549** onda bi string **S** trebao biti **S="-549"**. Štampati dobijeni string.
16. Definišimo **RIJEČ** kao string čiji prvi karakter mora biti veliko slovo, dok ostali karakteri mogu biti i mala i velika slova, kao i cifre. Kreirati algoritam i napisati pseudokod koji učitava string **STR** i koji provjerava i ispisuje da li dati string predstavlja **RIJEČ**.