ZG1 (20,21)

I

1. Učitati broj i ispitati je li prost. Koristiti pointere.
2. Učitati četiri broja i ispisati najmanji koristeći pointere.
3. Učitati 10 brojeva. Ispisati koliko je neparnih brojeva djeljivih sa 5, a koliko parnih djeljivih sa 7 koristeći pointere!
4. Učitati string koji sadrži manje od 50 karaktera i sva velika slova u tom stringu konvertovati u mala, a ostale karaktere koji nisu slova zamijeniti karakterom ‘#’.
5. Učitati evidenciju o nekoliko lica (ime, dužine manje od 15 karaktera, i ulog), a zatim štampati imena svih onih koji imaju ulog manji od prosječnog.
6. Napisati program koji za učitani prirodni broj računa sumu i proizvod svih prirodnih brojeva koji su manji od ili jednaki tom broju.
7. Napisati program koji računa srednju vrijednost niza realnih brojeva.
8. Napisati program koji provjerava da li je učitani prirodan broj N Hamming-ov broj. Prirodan broj je Hamming-ov ukoliko su mu jedini prosti činioci 2, 3 ili 5.
9. Napisati program koji će za unešeni prirodan broj, sve brojeve koji su manji od zadatog broja, a nisu prosti, smjestiti u novi niz. Program treba da sadrži i funkciju koja određuje da li je broj, koji joj se prosljeđuje kao argument, prost.
10. Napisati funkciju koja broji koliko se karaktera dva stringa, koji su joj argumenti, ne poklapa.
11. Napisati program koji sadrži funkciju koja izračunava najveći zajednički djelilac dva prirodna broja(NZD) koristeći Euklidov algoritam.

Napomena: U matematici, najveći zajednički djelilac (NZD) dva cijela broja je najveći pozitivan cijeli broj koji dijeli oba broja bez ostatka. Zadatak rješavamo pomoću Euklidovog algortima:

1) Ako je a=b, tada je NZD=a i to je kraj algoritma.

2) Od većeg broja oduzmemo manji i vraćamo se na prvi korak.

1. Kreirati program koji definiše niz cijelih brojeva X i koji vrši oduzimanje elemenata tog niza

korišćenjem pomoćne pokazivačke promjenljive. Prilagoditi program tako da eventualna izmjena broja elemenata elemenata niza u njegovoj inicijalizaciji ne zahtijeva intervenciju na programu.

1. Kreirati funckiju množenje tipa **void.** Za dva unijeta cijela broja funkcija računa njihov proizvod. Rezultat štampati iz **main** funkcije.
2. Učitati string dužine *n* karaktera i ispisati njegovu dužinu. Na početak tog stringa dodati

string “POCETAK”, a kraj tog stringa dodati string „ZAVRSETAK“. Ne koristiti funkciju strlen.

1. Napisati program koji ce učitati string od 10 karaktera, a na ekranu ispisati samo one karaktere koji predstavljaju velika slova. Na početni string nadovezati novi string od 5 karaktera i prikazati novi string. Ne koristiti ugradjene funkcije.
2. Napisati funkciju koja računa kvadrat binoma dva broja.
3. Napisati program koji sadrži funkciju za određivanje srednjeg po vrijednosti od 3 unijeta broja.
4. Napisati program za sortiranje niza preko pointera.
5. Napisati program koji kopira prvi string u drugi string. Ne koristiti ugradjene funkcije.
6. Napisati program koji računa sumu svih elemenata niza. Program realizovati pomoću pointera.

II

1. Učitati string iz tekstualnog fajla i provjeriti da li taj string predstavlja palindrom(npr. Ana voli Milovana)
2. Fajl se sastoji od slova a,b...z i znakova blanko. Napisati program koji određuje koliko ima riječi u kojima se bar jednom pojavljuje slovo a.
3. Učitati sadržaj fajla i ukloniti sve razmake.
4. Iz fajla učitati prvih 40 karaktera a zatim ukloniti karaktere ‘#’.
5. Sastaviti program koji kreira fajl koji se od polaznog razlikuje po tome sto su sva mala slova pretvorena u velika.
6. Napisati program koji upisuje sva slova i njihov ascii kod u novi fajl.
7. U fajlu je zadat jedan broj. Učitati ga i napisati program za pogađanje broja.
8. U svakom redu tekstualnog fajla upisana je jedna riječ, dužine do 20 karaktera. Riječi iz ovog fajla sortirati leksikografski i upisati ih u novi fajl.
9. Upisati piramidu proizvoljnih dimenzija u novi fajl.
10. U svakom redu u fajlu se nalazi po riječ, provjeriti koje riječi iz unijetog fajla sadrže jednak broj slova a i b, upisati ih u jednom redu u novom fajlu u formatu "riječ#riječ".
11. Unijeti niz dvocifrenih cijelih brojeva preko komandne linije i zamjeniti prvu i zadnju cifru svakog broja. Štampati nove brojeve.
12. Unijeti niz cijelih brojeva preko komandne linije i odstampati srednju cifru svakog broja (ako ima paran broj cifara odstampati dvije srednje cifre).

1. Unijeti katete trougla sa komandne linije. Zatim odstampati hipotenuzu. U slucaju da korisnik unese samo jednu racunati da je trougao jednakokraki, u slucaju da unese vise od dvije dati odgovarajuću grešku.
2. Unijeti niz cijelih brojeva sa komandne linije i odstampati medijan niza.
3. Preko komandne linije unijeti nekoliko riječi. Štampati koliko riječi sadrži slovo 'z' a zatim te riječi unijeti u novi fajl.
4. Unijeti niz cijelih brojeva preko komandne linije i provjeriti koji su od njih prosti i upisati ih u novi fajl.
5. Kao argument komande zadaju se tri parametra – riječ, slovo i broj. Tri parametra upisati u jedan string. Izmjeniti string tako da se umesto slova upisuje ASCII kod broja.
6. Kao argument preko komande linije daje se riječ. Napisati program koji od te riječi ispisuje riječ koja se dobija tako što se prvo slovo ponovi 1 put, drugo 2 puta itd. Upisati novu rijec u fajl.
7. Preko komandne linije se unosti niz cijelih brojeva. Sortirati niz u **strogo** rastući.
8. Preko komandne linije se unosti niz cijelih brojeva. Sortirati niz u **strogo** opadajući.
9. Unijeti string putem komandne linije za koga ce se dinamicki alocirati memorija. Onda realocirati memoriju kako bi se na kraj tog stringa nadovezao novi string proizvoljne duzine.