

**CS130/CS323: C/C++ programski jezik DOMAĆI ZADATAK 3.**

Prilikom slanja domaćeg zadatka neophodno je da ispunite sledeće:

* Subject mail-a mora biti CS323-DZbr (u slučaju kada šaljete domaći za ovu nedelju to je CS323-DZ03)
* U prilogu maila treba da se nalazi arhiviran projekat koji se ocenjuje, imenovan na sledeći način: CS323-DZbr-ImePrezimeBrojIndeksa. Na primer, CS323-DZ03-VeljkoGrkovic1234
* Potrebno je uraditi i printscreen koda programa sa rezultatom koji program daje
* Telo mail-a treba da ima pozdravnu poruku

**Molimo sve studente da se pridržavaju navedenog, inače zadaci neće biti pregledani i ocenjeni.**

**Svaki student radi po dva zadatka sa spiska**. Ostali zadaci mogu da posluže za vežbanje i pripremu ispita, ali ih ne šaljete profesoru ili asistentima na pregledavanje.

Student bira zadatke po sledećim formulama:

***Zadatak 1: Broj indeksa % 15 + 1 (Npr. 2378 % 15 + 1 = 9 – Student radi 9. zadatak).***

***Zadatak 2: Broj indeksa % 15 + 16 (Npr. 2378 % 15 + 16 = 24 – Student radi 24. zadatak).***

***Grupa zadataka* 1:**

1. Kreirati metod koji ce promeniti zadati niz brojeva od 10 celih brojeva promeniti na sledećinačin. Ako je indeks elementa deljiv sa 3, broj na toj poziciji će uvećati dva puta, ako je indeks elementa deljiv sa 5 element na toj umanjiti na pola, preostale elemente ne treba promeniti.
2. Kreirati metod koji ce promeniti zadati niz brojeva od 20 celih brojeva promeniti na sledećinačin. Ako je indeks elementa deljiv sa 3, broj na toj poziciji će umanjiti za 2, ako je indeks elementa deljiv sa 5 element na toj poziciji će kvadrirati, elementima na parnim pozicijma treba dodati vrednost 10, a preostale elemente ne treba promeniti.
3. Slučajnim izborom formirati niz od 20 elemenata iz intervala 0-9.Izračunati sumu parnihelemenata, prebrojati koliko je elemenata jednako nuli, pa zatim prikazati niz i dobijene rezultate.
4. Napisati program koji formira niz od 1000 slučajno odabranih elemenata iz intervala od 0 9999, i određuje najveći element niza.
5. Napisati program u C-u koji učitava niz realnih brojeva od n elemenata i realan broj x.Napisati funkciju **PoslednjaPojava** koja vraća najveći indeks elemenata niza koji je jednak broju x, a ako ga nema vraća -1. Iz glavnog programa pozvati funkciju **PoslednjaPojava**.
6. Napisati program koji formira niz od 1000 slučajno odabranih elemenata iz intervala od 0 9999. Prikazati niz i prebrojati koliko je elemenata u nizu jednako nuli.
7. Napisati program u C-u koji učitava niz realnih brojeva od n elemenata. Napisati funkciju**PozitivniOd** koja transformiše učitani niz tako što svaki element niza zamenjuje brojem pozitivnih elemenata niza od posmatranog elementa do kraja (ukljucujuci i posmatrani element). U glavnom delu programa ispisati dobijeni niz.
8. Slučajnim izborom formirati niz od 20 elemenata iz intervala 0-100. Odrediti maksimalniparni element niza i minimalni neparni element niza.
9. Formirati slučajnim izborom niz a[] od 100 elemenata iz intervala od 1 - 200. Iz takvog nizaizdvojiti podniz b[] koji će sadržati samo dvocifrene brojeve niza a[].
10. Napisati program koji formira nizo od n elemenata slučajnim izborom brojeva od 0 do 99. Izračunati sumu parnih elemenata većih od 60.
11. Formirati dva niza A[] i B[] od m i n elemenata. Trećiniz C[] se automatski formira od dvauneta niza tako što svi parni elementi nizova A i B postaju elementi trećeg niza. Prikazati elemente sva tri niza.
12. Formirati niz A[] od n slučajno odabranih elemenata iz intervala od 0-100. Takav nizpodeli na dva niza: nizB[] sadrži elemente manje od 50, niz C[] sadrži ostale elemente.
13. Napisati funkciju za izračunavanje aritmetičke sredine niza i testirati je. Aritmetickasredina je povratna vrednost funkcije.
14. Napisati program koji učitava brojeve n i m sa standardnog ulaza, zatim učitava n redovai m kolona dvodimenzionalnog niza i ispisuje sumu svih kolona dvodimenzionalnog niza.
15. Napisati program kojim se učitava ceo broj n (n<20), dva celobrojna niza a i b dužine n ipotom formira niz c naizmenicnim mešanjem elemenata nizova a i b. Primer ulaza: n=5

a: 1 2 5 6 9 b: 6 3 1 8 0 Rezultat: c: 1 6 2 3 5 1 6 8 9 0

***Grupa zadataka 2:***

1. Napisati funkciju koja čita liniju po liniju iz konzole i za svaku učitanu liniju ispisuje brojznakova koji su različiti od praznine. Maksimalna dužina linije koja se čita neka bude 20 karaktera.
2. Napisati program koji učitava dva stringa koji predstavljaju cele brojeve NPR "123", "456".Od ova dva uneta stringa formira sledeće cele brojeve a="123", b="456", c="123456".
3. Napisati funkciju koja dati string s skraćuje na podstring od prvih p znakova. Korisnikunosi p.
4. Napisati program koji u stringu s određuje broj pojavljivanja znaka c (s i c učitavamo satastature). Koristiti funkciju u kojoj se računa broj poljavljivanja znaka i rezultat se vraca glavnom programu.
5. Napisati funkciju koja računa vrednost stringa koji se sastoji iz vašeg imena i prezimena.

Funkcija treba da sabere ASCII vrednosti svakog karaktera u vašem imenu i prezimenu. Funkcija vraća broj.

1. Napisati funkciju void **zameni**(char s[]) koja u stringu s maksimalne duzine 20 karaktera zamenjuje svako pojavljivanje slova **a** slovom **b**. Na primer: niska d**a**n se zamenjuje niskom d**b**n. U glavnom program učitati nisku sa ulaza i testirati funkciju.
2. Napisati funkciju kojom se ispituje da Ii je string palindrom. String je palindrom ako se istočita s leva kao s desna.Testirati funkciju u glavnom programu.
3. Napisati program koji umeće string s2 u string s1 počev od pozicije p. Korisnik unosi p.
4. Napisati funkciju kojom se ulazni string s ispisuje sa k znakova u vrsti (izuzev poslednjevrste koja može imati manje od k znakova).

Ulaz: aaaaaaaaaaaaaaaaaaaaaaa k = 5

Izlaz:

aaaaa aaaaa aaaaa aaaaa

…

1. Napisati program koji učitava dva stringa, određuje broj pojavljivanja znaka 'a' izražen uprocentima. Prikazati rezultate i učitane stringove.
2. Napisati program koji odredjuje koliko ima karaktera +, \*, /, - u unetom stringu.
3. Napisati funkciju koja izbacuje sve nule sa kraja stringa.
4. Napisati funkciju int **anagrami**(char s1[], char s2[]) kojom se proverava da li su dve zadate niske anagrami (za dve niske kazemo da su anagrami ako se od slova prve niske moze formirati druga niska).

Na primer: ortoped i torpedo

U glavnom programu ucitati dve niske maksimalne duzine 20 i testirati rad funkcije

1. Napisati funkciju kojom se u stringu s izbacuje prvo pojavljivanje stringa s1.
2. Odrediti broj reči u stringu koje pocinju sa 'B'.