



## CS203 - DOMAĆI ZADATAK 04.

Prilikom slanja domaćeg zadatka svom asistentu neophodno je da ispunite sledeće:

- Subject mail-a mora biti CS203-DZbr. Za ovaj domaći zadatak - CS203-DZ01
- Sve fajlove, koji su deo rešenja zadataka, arhivirati (zip)
- **Obavezno kopirati tekst zadatka kao komentar u kodu resenja**
- U prilogu maila treba da se nalazi arhiva projekta koji se ocenjuje, imenovana na sledeći način: CS203-DZbr-BrojIndeksa-ImePrezime.

*Na primer, CS203-DZ01-9999-VeljkoGrkovic.zip*

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

Studenti iz Beograda zadatke na ocenjivanje šalju mail-om na adresu [lazar.mrkela@metropolitan.ac.rs](mailto:lazar.mrkela@metropolitan.ac.rs)

Svaki student radi jedan zadatak sa spiska. Ostali zadaci mogu da posluže za vežbanje i pripremu ispita, ali ih ne šalžete na pregled asistentima.

Redni broj zadatak dobijate tako što vaš broj indeksa podelite sa 22, a dobijeni ostatak pri deljenju uvećate za 1. Na primer:

**Broj indeksa 2378**

**$2378 \% 22 + 1 = 3$  – student radi 3. zadatak.**

1. Napisati program koji korišćenjem rekurzivne funkcije računa sumu:

$$S = 1 + \frac{x}{1!} + \frac{x^2}{2!} + \frac{x^3}{3!} + \dots + \frac{x^N}{N!}$$

2. Napisati program koji korišćenjem rekurzivne funkcije računa sumu:

$$S = 1 - \frac{x}{1!} + \frac{x^2}{2!} - \frac{x^3}{3!} + \dots + (-1)^N \frac{x^N}{N!}$$

3. Za dati prirodan broj pomoću rekurzivne funkcije ispisati broj koji ima iste cifre, ali u obrnutom poretku.
4. Napisati rekurzivnu funkciju koja određuje da li je zbir cifara nekog celog broja paran.
5. Napisati rekurzivnu funkciju koja nalazi najveći od datih **N** brojeva.
6. Dat je **niz A** od **N** elementa i **broj X**. Napisati rekurzivnu funkciju koja vraća indeks elementa niza koji je jednak X, u suprotnom vraća nulu ako ne postoji element koji je jednak X.
7. Napisati rekurzivnu funkciju za računanje binomnog koeficijenta **C(m,n)**, gde je  $0 \leq m \leq n$ , po formuli:

$$C(0,n) = C(n,n) = 1;$$

$$C(m,n) = C(m,n-1) + C(m-1,n-1), \text{ za } 0 < m < n;$$

8. Napisati program u C-u koji učitava niz realnih brojeva od **n** elemenata i realan broj **x**. Napisati rekurzivnu 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**.
9. Napisati program u C-u koji učitava niz realnih brojeva od **n** elemenata. Napisati rekurzivnu funkciju **IspisKraja** koja niz brojeva ispisuje od poslednjeg ka prvom. Iz glavnog programa pozvati funkciju **IspisKraja**.
10. Napisati program u C-u koji sadrži rekurzivnu funkciju **verizni** koja za dato **n** računa vrednost verižnog razlomka, npr.:

$$Verizni(1) = 1 + \frac{1}{1}; \quad Verizni(2) = 1 + \frac{1}{1 + \frac{1}{1}}; \quad Verizni(3) = 1 + \frac{1}{1 + \frac{1}{1 + \frac{1}{1}}}; \quad \dots$$

U glavnom programu učitati **k** i ispisati vrednost f-je **verizni** za **k**.

11. Napisati program u C-u koji učitava niz nula i jedinica koji predstavlja binarni zapis broja (od cifre najveće do cifre najmanje težine). Napisati rekurzivnu funkciju **Dekadni** koja dobija niz binarnih cifara i vraća dekadnu vrednost čiji je binarni zapis prosleđen. Iz glavnog programa pozvati funkciju **Dekadni**.

12. Neka je data matrica **A** dimenzija **n x n** i neka ta matrica predstavlja neku relaciju **r**. Napisati rekurzivnu funkciju koja proverava da li je relacija simetrična.
13. Neka je data matrica **A** dimenzija **n x n** i neka ta matrica predstavlja neku relaciju **r**. Napisati rekurzivnu funkciju koja proverava da li je relacija refleksivna.
14. Sastaviti rekurzivnu funkciju koja sa prototipom **int cifra(int d, int n)** koja ispituje da li se cifra **d** pojavljuje u zapisu broja **n**.
15. Sastaviti rekurzivnu funkciju koja računa NZD dva broja.
16. Napisati rekurzivnu funkciju koja računa broj cifara celog broja **X**, koji je njen argument.
17. Napisati rekurzivnu funkciju koja leksikografski poredi dva stringa.
18. Ukoliko Paskalov trougao posmatramo kao trougaonu matricu, napisati rekurzivnu funkciju koja računa vrednost elementa Paskalovog trougla sa koordinatama **(i, j)**.
19. Napisati rekurzivnu funkciju koja računa i ispisuje u datoteku permutacije.txt sve permutacije dužine 5 brojeva od 1 do 5.  
Na primer, {1,2,3,4,5} bi bila jedna permutacija  
                  {5,4,2,1,3} bi bila druga permutacija
20. Napisati rekurzivnu funkciju koja ispituje da li je uneti string palindrom. Testirati rad funkcije u glavnom programu.
21. (Permutacije stringova) Napisati iterativni i rekurzivni metod koji štampa sve permutacije stringova. Na primer, za string **abc**, biće odštampano: **abc, acb, bac, bca, cab, cba**.
22. Napisati rekurzivnu funkciju koja izračunava skalarni proizvod dva vektora celih brojeva. Napisati program koji testira ovu funkciju za nizove (vektore) koji se unose sa standardnog ulaza. Prvo treba uneti dimenziju nizova, a zatim i njihove elemente. Na standardni izlaz ispisati skalarni proizvod unetih nizova. Pretpostaviti da nizovi neće imati više od 256 elemenata.