



CS103 - DOMAĆI ZADATAK 01.

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

- Subject mail-a mora biti CS103-DZbr. Za ovaj domaći zadatak - CS103-DZ01
- Sve fajlove, koji su deo rešenja zadataka, arhivirati (zip, rar, ...)
- Poželjno je uraditi i printscreen koda pre pokretanja programa
- U prilogu maila treba da se nalazi arhiva projekta koji se ocenjuje, imenovana na sledeći način: CS103-DZbr-BrojIndeksa-ImePrezime. *Na primer, CS103-DZ01-9999-VeljkoGrkovic.zip*
- Telo mail-a treba da ima pozdravnu poruku

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

Studenti iz Beograda i online studenti zadatke na ocenjivanje šalju mail-om na adresu

lazar.mrkela@metropolitan.ac.rs

Studenti iz Niša zadatke na ocenjivanje šalju mail-om na adresu

nikola.dimitrijevic@metropolitan.ac.rs

Rok za slanje domaćih zadataka:

- **Tradicionalni studenti: sedam dana od dana održavanja vežbe na koju se odnosi domaći zadatak. Nakon toga, broj poena se umanjuje za 50%. Krajnji rok za slanje rešenja je deset dana pre termina ispita u kome želite da polažete ispit.**
- **Studenti na online nastavi, kao i studenti kojima je predmet prenet u narednu godinu ili je diferencijalni: najkasnije deset dana pre termina ispita u kome želite da polažete ispit.**

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 20, a dobijeni ostatak pri deljenju uvećate za 1. Na primer:

Broj indeksa 2378

$2378 \% 20 + 1 = 19$ – student radi 19. zadatak.

1. Napisati funkciju koja za uneti ceo broj N računa najveći prost broj koji je njegov delilac (faktor). Primer: za broj 13195 najveći prost faktor je 29.
2. Napisati program koji računa najveći palindrom broj koji je nastao kao proizvod dva dvocifrena broja. Primer: $9009 = 91 \times 99$
3. Napisati program koji generiše sve brojeve od 1 do 1 000 000 koji su palindromi i u dekadnom i u binarnom zapisu. Primer: $585 = 1001001001_2$ (binary)
4. Napisati program koji generiše sve brojeve od 1 do 1 000 000 koji su palindromi i u dekadnom i u oktalnom zapisu.
5. Napisati program koji pronalazi sve trocifrene brojeve koji su cirkularno prosti brojevi. Broj je cirkularno prost ako su prosti svi brojevi koji mogu da nastanu rotacijom cifara. Primer 197 je cirkularno prost, jer su prosti i brojevi 197, 971, 719.
6. Napraviti funkciju koja vrsi konverziju broja u dekadnom zapisu u heksadecimalni. Napraviti niz koji sadrži 15 brojeva u dekadnom zapisu i prethodnom funkcijom transformisati ih u heksadecimalni broj.
7. Napraviti funkciju koja vrsi konverziju broja iz heksadecimalnog zapisa u dekadni, potom kreirati niz od 15 elemenata i prethodnom funkcijom konvertovati elemente niza.
8. Napisati funkciju kojom se određuje broj (koliko ima) različitih elemenata celobrojnog niza $A(N)$.
9. Napisati program koji unosi broj sa standardnog ulaza, formira broj sa ciframa u obrnutom poretku i ispisuje ga na standardni izlaz.
10. Za proizvoljni niz celih brojeva čija je dužina N ispitati da li je rastući. Niz je rastući ako vazi da je svaki naredni član niza veći ili jednak od prethodnog.
11. Napisati program kojim se štampaju svi trocifreni brojevi koji imaju osobinu da su deljivi sa brojem koji se dobija izbacivanjem srednje cifre tog istog broja.
12. Napisati program koji proverava da li je matrica A simetrična.
13. Napisati program koji transponuje matricu A unetu na standardnom ulazu.
14. Napisati program kojim se pronalaze svi brojevi od 100 do 10 000 koji imaju osobinu da se njihov zbir sa njihovim inverzom daje broj koji je palindrom. Primer: 47, obrnut broj 74, $47 + 74 = 121$. 121 je palindrom. Dakle, broj 47 ispunjava tražena svojstva.

15. Napisati funkciju koja pronalazi sve brojeve od 100 do 100 000 koji imaju tu osobinu da su deljivi brojem koji nastane rotacijom tog broja za jednu cifru. Primer: Neka je naš broj 142857, nakon rotacije dobijeni broj je 714285. 142857 je delilac novonastalog broja. $714285=5 \times 142857$,
16. (Najveći blok) Za zadatu kvadratnu matricu čiji su elementi samo 0 i 1, napisati program koji pronalazi najveću kvadratnu podmatricu čiji su elementi samo jedinice. Korisnik treba da unese broj vrsta matrice, i matricu (vrstu po vrstu). Program treba kao rezultat da prikaže lokaciju prvog elementa najveće kvadratne podmatrice, kao i broj vrsta najveće podmatrice. Program treba da implementira i da koristi sledeći metod za određivanje najveće kvadratne podmatrice:
`public static int[] findLargestBlock(int[][] m)`
17. Napisati funkciju `int suma_delilaca(int n)` koja određuje sumu pravih delilaca celog broja n (tj. sumu svih delilaca broja n, ne računajući sam broj n). Napisati potom program koji sa standardnog ulaza učitava ceo broj $n > 1$ i ispisuje na standardni izlaz sumu njegovih delilaca, ne računajući sam taj broj, korišćenjem implementirane funkcije. Ako n nije veće od 1, program treba da na standardni izlaz za grešku ispiše -1.
18. (Zajednički prefiks) Napisati metod koji vraća najduži zajednički prefiks dva stringa. Na primer, najduži zajednički prefiks stringova `distance` i `disinfection` je `dis`. Zaglavlje metoda treba da bude:
`public static String prefix(String s1, String s2)`
Ukoliko dva stringa nemaju zajednički prefiks, rezultat treba da bude prazan string. Napisati i glavni main metod pomoću kojeg se učitavaju stringovi iz konzole, poziva metod `prefix`, a zatim i štampa najduži zajednički prefiks dva stringa.
19. Napisati program kojim se štampaju svi brojevi od 1000 do 10 000 koji imaju osobinu da su deljivi sa brojem koji se dobija izbacivanjem cifara desetica i hiljada iz tog istog broja.
20. Napisati program koji računa najveći palindrom broj koji je nastao kao proizvod dva trocifrena broja.