

Fundamentele programării

Student:

Grupa:

Data:

Ora:

Sub. I	Sub. II	Sub. III*	Sub. IV*
1p	2p	2p	2p

Bilet nr. 101

Implementați și testați următoarele funcții conform cerințelor precizate:

- (I) Realizați un program care citește trei numere întregi de la tastatură și afișează: **TOATE NUMERELE SUNT PARE** sau **NU TOATE NUMERELE SUNT PARE**, după cum este cazul.

- (II) `double suma(int n){...}`

returnează 0 dacă $n \leq 0$, altfel returnează suma primilor n termeni ai seriei

$$\frac{1}{1^6} - \frac{1}{2^6} + \frac{1}{3^6} - \frac{1}{4^6} + \frac{1}{5^6} - \frac{1}{6^6} + \dots = \frac{62}{64} \cdot \frac{\pi^6}{945};$$

Verificați printr-un program egalitatea dată.

- (III) `void afiseazaTablou(int tab[], int n){...}`

afișează primele n elemente ale tabloului `tab` fără să le schimbe ordinea, dar sărind peste cele egale cu zero și schimbând eventual semnul numărului afișat, astfel încât pe monitor să nu existe trei numere consecutive de același semn.

Exemplu:

`tab:` 5 2 2 -6 -2 3 4 0 5 7 -3 0 -5 -2 4 4

`monitor:` 5 2 -2 -6 2 3 -4 5 7 -3 -5 2 4 -4

Se vor efectua cât mai puține schimbări de semn.

- (IV) `double maxim(int n){...}`

returnează 0 dacă $n \leq 0$, altfel returnează valoarea lui ω_n definit astfel

$$\omega_n = \max \left\{ \frac{1 \cdot (n-1)}{n^2 + 1^2}, \frac{2 \cdot (n-2)}{n^2 + 2^2}, \frac{3 \cdot (n-3)}{n^2 + 3^2}, \dots, \frac{(n-1) \cdot 1}{n^2 + (n-1)^2} \right\}.$$

Verificați printr-un program că

$$\lim_{n \rightarrow \infty} \omega_n = \frac{\sqrt{2} - 1}{2}.$$