

Programe simple în C/C++



Obiective

Specificarea, proiectarea și implementarea problemelor simple în C/C++.



Aspecte teoretice

Aspecte generale ale limbajului C++.

Aplicații în C++ - versiune structurată și versiune modulară.



Termen de predare

Live - Problemele trebuie rezolvate în cadrul laboratorului 1.



Cerințe

Scrieți 2 variante ale aceluiași program în C++ pentru a rezolva una dintre următoarele probleme.

- primul program trebuie să fie o aplicație structurată într-un singur fișier
- al doilea program trebuie să fie o aplicație modulară; aplicația trebuie să conțină minimum 5 fișiere

Evaluare:

- procedurală (un singur fișier sursă) - 6p
 - citire date de intrare - 1p
 - afișare rezultate - 1p
 - rezolvare propriu-zisă a cerinței - 2p + teste 2p
- modulară (5 fișiere: problem.h, problem.cpp, tests.h, tests.cpp, app.cpp) – 4p

Probleme tip A

Să se rezolve una dintre problemele de mai jos fără a folosi șiruri.

1. Fiind dat un număr natural n , să se determine numerele naturale prime p_1 și p_2 astfel încât $n = p_1 + p_2$.
2. Fiind dat un număr natural n , să se determine cel mai mic număr natural m format din aceleași cifre cu n . De ex. pentru $n=3658$, $m=3568$.
3. Fiind dat un număr natural n , să se determine cel mai mare număr natural m format din aceleași cifre cu n . De ex. pentru $n=3658$, $m=8653$.
4. Fiind dat un număr natural n , să se determine cel mai mare număr prim mai mic decât n . Dacă nu există un astfel de număr, se va afișa un mesaj.
5. Fiind dat un număr natural n , să se determine primele două numere prime gemene mai mari decât n . Două numere prime p și q sunt gemene dacă $q - p = 2$.
6. Fiind dat un număr natural n , să se determine cel mai mic număr m din șirul Fibonacci (definit prin $f[0]=f[1]=1$, $f[n]=f[n-1]+f[n-2]$, pentru $n>2$) mai mare decât n .
7. Fiind dat un număr natural n , să se determine produsul factorilor proprii ai lui n .
8. Fiind dat un număr natural n , să se determine palindromul lui n .

Probleme tip B (bonus 1p)

1. Înmulțirea *à la russe* a două numere naturale.
2. Partea întregă a unui număr întreg. Propuneți o metodă bazată pe operații de adunare, scădere și comparare pentru determinarea părții întregi a unui număr real. Partea întregă a unui număr real este definită ca fiind cel mai mare număr întreg mai mic decât numărul real. De exemplu, partea întregă a lui 2.4 este 2, iar a lui -2.4 este -3.
3. Calculul unei sume. Fie n un număr natural, x o valoare reală din $(0, 1)$ și $\varepsilon > 0$ o valoare reală pozitivă. Descrieți un algoritm pentru calculul sumelor $\sum_{i=1}^n \frac{(-1)^i x^{2i}}{(2i)!}$ și $\sum_{i=1}^{\infty} \frac{(-1)^i x^{2i}}{(2i)!}$ (calculul aproximativ cu precizia ε).
4. Elemente din șir. Să se afișeze primele N elemente și să se aproximeze (cu precizia ε) limitele șirurilor:
 - (a) $x_n = (1 + 1/n)^n$;
 - (b) $x_1 = a > 0$, $x_n = (x_{n-1} + a/x_{n-1})/2$;
 - (c) $x_n = f_{n+1}/f_n$, $f_1 = f_2 = 1$, $f_n = f_{n-1} + f_{n-2}$.