# Probleme C++ Clasa a 12-a - Rezolvări și Explicații

## 1. Inserare într-un vector ordonat crescător (fișier)

Enunț:

Fișierul sir.in conține pe prima linie un număr natural n (2 < n < 100), iar pe a doua linie sunt n numere naturale ordonate crescător. Pe linia a treia se află un număr natural x. Să se scrie un program care memorează șirul într-un vector și apoi inserează pe x astfel încât vectorul să rămână ordonat crescător. Se va afișa apoi vectorul în fișierul sir.out.

Exemplu:

sir.in  
6  
2 4 4 7 8 12  
6  
sir.out  
2 4 4 6 7 8 12

Explicație:

Citim numerele într-un vector, apoi găsim poziția unde trebuie inserat x, astfel încât șirul să rămână ordonat. Parcurgem vectorul și inserăm x la poziția potrivită.

Cod C++:

#include <iostream>  
#include <fstream>  
using namespace std;  
  
int main() {  
 ifstream f("sir.in");  
 ofstream g("sir.out");  
  
 int v[101], n, x, i, poz = 0;  
 f >> n;  
 for (i = 0; i < n; i++) f >> v[i];  
 f >> x;  
 while (poz < n && v[poz] <= x) poz++;  
 for (i = n; i > poz; i--) v[i] = v[i - 1];  
 v[poz] = x;  
 for (i = 0; i <= n; i++) g << v[i] << ' ';  
 return 0;  
}

## 2. Valoarea maximă și cea imediat mai mică

Enunț:

Fișierul numere.in conține pe prima linie un număr n, iar pe linia a doua n numere naturale. Se cere valoarea maximă și valoarea imediat mai mică decât maximul.

Exemplu:

numere.in  
6  
88 123456 7 123456 992292 835  
Se va afișa:  
Valoare maxima: 992292  
Valoare imediat mai mica: 123456

Explicație:

Parcurgem vectorul, memorăm maximul și apoi parcurgem din nou pentru a găsi cel mai mare număr < maxim.

Cod C++:

#include <iostream>  
#include <fstream>  
using namespace std;  
  
int main() {  
 ifstream f("numere.in");  
 int n, x, max1 = 0, max2 = 0;  
 f >> n;  
 int v[n];  
 for (int i = 0; i < n; i++) {  
 f >> v[i];  
 if (v[i] > max1) max1 = v[i];  
 }  
 for (int i = 0; i < n; i++) {  
 if (v[i] < max1 && v[i] > max2) max2 = v[i];  
 }  
 cout << "Valoare maxima: " << max1 << "\n";  
 cout << "Valoare imediat mai mica: " << max2 << "\n";  
 return 0;  
}