

POO – C++ - Laborator 2

Definirea si utilizarea tipurilor de data structură in C și C++

1. (ANSI C) Creați un tip de date structură numit *student* care să conțină: *număr matricol* de tip *int*, *nume* de tip șir de caractere *char[]*, *gen* de tip *char*, *nota* de tip *float* mobilă simplă precizie *float*, doi pointeri pe funcții pentru citire date *void (read*)(student*st)* și unul pentru afisare date *void (write*)(student*st)*.

Citiți de la tastatura (folosind *cin*) numărul de studenți și alocați dinamic memorie pentru un vector de studenți (folosiți *v = new student[n]*).

Definiți două funcții: *void ReadData(student*st){...}* și *void WriteData(student*st){...}* care citesc și afisează membrii unei variabile de tip structură *student*. Parcurgeți vectorul cu elemente de tip *student* și inițializați pointerii la funcții *void (read*)(student*st)* și *void (write*)(student*st)* cu funcțiile *void ReadData(student*st)* și *void WriteData(student*st)*.

Parcurgeți vectorul cu elemente de tip *student* și apelați funcțiile: *v[i].read(&v[i])* și *v[i].write(&v[i])* pentru citirea datelor și respectiv pentru afisarea datelor.

2. (C++) Declarați două funcții membre ale structurii *student* *void Read(void)* și *void Write(void)* și definiți aceste funcții în afara structurii: *void student::Read(void){...}* și *void student::Write(void){...}*.

Parcurgeți vectorul cu elemente de tip *student* și apelați funcțiile: *v[i].Read()* și *v[i].Write()* pentru citirea datelor și respectiv pentru afisarea datelor.

Analizați și discutați cele două abordări.

Tema

Creați un tip de date structură numit *catalog* care să conțină numărul de studenți, un pointer pe șiruri de caractere (vector de șiruri alocate dinamic), două funcții membre pentru sortare alfabetică și sortare după lungime, două funcții membre pentru citirea și afisarea datelor. Definiți o funcție de creare a unui catalog (aloca memorie pentru un tip de date catalog, apelează funcția de citire a datelor) și o funcție de distrugere a unei variabile de tip catalog (dealoca spațiul de memorie ocupat). Scrieți un program care să utilizeze un astfel de tip de date.

obs. nu uitați dealocarea zonelor de memorie (folosiți *delete []v*)