

PROIECT DE LECȚIE

Școala: Liceul Teoretic “Nicolae Bălcescu” Cluj-Napoca

Profesor: Călăuz Răzvan

Clasa: 10 MI 2

Profil/Specializare: Matematică-Informatică

Data: 21.03.2025

Disciplina: Informatică

Unitatea de învățare: Tipul de date struct

Titlul lecției: Tipul de date înregistrare

Tipul lecției: transmitere și asimilare de noi cunoștințe

Competențe generale: Identificarea datelor care intervin într-o problemă și aplicarea algoritmilor fundamentali de prelucrare a acestora

Competențe specifice:

1. Utilizarea șirurilor de caractere și a structurilor de date neomogene în modelarea unor situații problemă;
2. Implementarea unor algoritmi de prelucrare a șirurilor de caractere și a structurilor neomogene.

Obiective operaționale: La finalul lecției, elevul va fi capabil să:

- Explice importanța tipului de date înregistrare;
- Declare și inițializeze înregistrări în limbajul C++;
- Acceseze și folosească înregistrările într-o problemă practică;
- Implementeze operațiile de adăugare, citire și scriere din fișier pe o înregistrare.

Desfășurarea lecției:

Evenimentele lecției	Activitatea din lecție	Strategia didactică și evaluarea
Captarea atenției	Punerea unor întrebări generale/introductive: La ce vă gândiți când auziți cuvântul „înregistrare”? Explicare scurtă a conceptului de înregistrare și a utilizării acestui tip de date.	Conversația
Reactualizarea cunoștințelor anterior însușite	Ce facem când vrem să gestionăm o listă de elevi? Cum o reprezentăm în calculator? Cum declarăm un vector/tablou unidimensional?	Evaluarea frontală
Informarea elevilor asupra obiectivelor urmărite	Titlul lecției: Gestionarea datelor folosind înregistrări	Conversația
Prezentare de material nou	Noțiuni teoretice (prin intermediul unei probleme): Problema: Gestionarea unei liste de produse într-un magazin (nume, preț, cantitate). <ul style="list-style-type: none"> • Definirea înregistrării: <pre>struct Produs { string nume; float pret; int cantitate; }; const int MAX_PRODUSE = 100; Produs magazin[MAX_PRODUSE]; int nrProduse = 0;</pre>	Explicația

	<ul style="list-style-type: none"> • Adăugarea unui produs: <pre>void adaugaProduce(string nume, float pret, int cantitate) { if (nrProduce < MAX_PRODUSE) { magazin[nrProduce].nume = nume; magazin[nrProduce].pret = pret; magazin[nrProduce].cantitate = cantitate; nrProduce++; } else { cout << "Nu se mai pot adăuga produse!" << endl; } }</pre> <ul style="list-style-type: none"> • Afișarea produselor: <pre>void afiseazaProduce() { for (int i = 0; i < nrProduce; i++) { cout << magazin[i].nume << " - " << magazin[i].pret << " RON - " << magazin[i].cantitate << " buc." << endl; } }</pre> <ul style="list-style-type: none"> • Scrierea produselor în fișier: <pre>void salveazaInFisier() { ofstream fisier("produse.txt"); for (int i = 0; i < nrProduce; i++) { fisier << magazin[i].nume << ", " << magazin[i].pret << ", " << magazin[i].cantitate << endl; } fisier.close(); }</pre> <ul style="list-style-type: none"> • Actualizarea unui produs: <pre>void actualizeazaProduce(int index, string numeNou, float pretNou, int cantitateNoua) { if (index >= 0 && index < nrProduce) { magazin[index].nume = numeNou; magazin[index].pret = pretNou; magazin[index].cantitate = cantitateNoua; } else { cout << "Index invalid!" << endl; } }</pre> <ul style="list-style-type: none"> • Ștergerea unui produs: <pre>void stergeProduce(int index) { if (index >= 0 && index < nrProduce) { for (int i = index; i < nrProduce - 1; i++) { magazin[i] = magazin[i + 1]; } nrProduce--; } else { cout << "Index invalid!" << endl; } }</pre>	
Dirijarea învățării	<p>Activitate practică: Se consideră o listă de produse, citite din fișier. Se cere să:</p> <ul style="list-style-type: none"> • Se calculeze totalul produselor din magazin: <pre>int calculeazaTotalCantitati() { int total = 0; for (int i = 0; i < nrProduce; i++) { total += magazin[i].cantitate; } }</pre>	

	<pre> } return total; } • Se afișeze toate produsele care încep cu litera “c”:</pre> <pre> void afiseazaProduseCuLiteraC() { for (int i = 0; i < nrProduse; i++) { if (magazin[i].nume[0] == 'C' magazin[i].nume[0] == 'c') { cout << magazin[i].nume << " - " << magazin[i].pret << " RON - " << magazin[i].cantitate << " buc." << endl; } } } }</pre>	
Asigurarea conexiunii inverse	Ce trebuie schimbat în structura înregistrării dacă vrem să reținem și o listă de ingrediente pentru un produs?	Calculatorul, învățarea prin descoperire, problematizarea, dezbateră
Asigurarea reținerii	Se consideră o listă de mașini, citite din fișier. O mașină are nume, marcă, culoare, an și preț. Să se afișeze toate acele mașini de marca Toyota care au culoarea roșie.	Tema pentru acasă, evaluare independentă