

C, C++ și C#



DEAF YOUNG
CODE

Număr proiect: 2023-2-IT03-KA220-YOU-000179130



Co-funded by
the European Union

Obiective de învățare



Până la finalul acestei lecții, elevii vor putea să:

- **A înțelege ce sunt C, C++, și C# .**
- **A cunoaște principalele diferențe dintre aceste limbi.**
- **A citi și a înțelege exemple simple în fiecare limbă.**
- **A recunoaște conceptele de bază ale programării: variabile, bucle și funcții.**
- **A Conecta conceptele de cod la logica vizuală (diagrame de flux, diagrame).**



Video



[Click aici](#)
[pentru video](#)



Ce este C?

Definition:

- C este un limbaj de programare de nivel înalt creat în anii 1970.
- Folosit pentru a scrie programe rapide și a controla hardware-ul calculatorului.

Caracteristici:

- Programare procedurală (pas cu pas)
- Simplu și rapid
- Folosite în software-ul de sistem (de exemplu, sisteme de operare)



Example (C – Print Hello World):



```
#include <stdio.h>
```

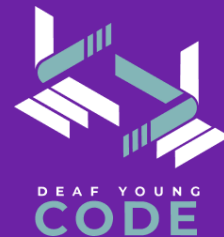
```
int main() {  
    printf("Hello, World!");  
    return 0;  
}
```

Visual Tip: Use arrows to show step-by-step execution:

include → main → printf → output



Ce este C++?



Definiție:

- C++ este o extensie a lui C.
- Adaugă caracteristici de programare orientată pe obiecte (OOP): clase, obiecte.

Caracteristici:

- Poate face tot ce poate face C
- Suportă obiecte și clase
- Folosit pe scară largă în jocuri, aplicații și software grafic



Exemplu (C++ – Using a class):



```
#include <iostream>

using namespace std;

class Car {
    public:
        void start() {
            cout << "Car started!" << endl;
        }
};
```

**Visual Tip: Show class → object → method →
output with boxes/arrows**



Ce este C# (C Sharp)?



Definiție:

- C# este un limbaj modern creat de Microsoft.
- Conceput pentru aplicații Windows, web și jocuri (Unity).

Caracteristici:

- Orientat pe obiecte
- Ușor de folosit, sintaxă modernă
- Funcționează cu .NET framework



Example (C# – Print Hello World):



```
using System;

class Program {
    static void Main() {
        Console.WriteLine("Hello, World!");
    }
}
```

Visual Tip: Highlight similarities with C++
(Main function, print/output)



Diferențe cheie

Feature	C	C++	C#
An	1972	1983	2000
Paradigmă	Procedural	Procedural + OOP	Object-oriented
Utilizare principală	Sisteme	Apps, Games	Apps, Web, Unity
Complexitate	Simplu	Mediu	Modern, ușor



Exemplu simplu de comparație

Task: Print “Hello, World!”



```
printf("Hello, World!");
```



```
cout << "Hello, World!";
```



```
Console.WriteLine("Hello, World!");
```



Activitate practică



Activitate:

1. Scrie un program mic în orice limbaj care îți tipărește numele.
2. Desenează o diagramă de flux a ceea ce se întâmplă pas cu pas.

1.

Task: Scrie un program în oricare dintre cele trei limbaje care îți imprimă numele.

C: Use `printf("YourName");`

C++: Use `cout << "YourName";`

C#: Use `Console.WriteLine("YourName");`



Activitate practică

2. Desenează o diagramă de flux: Arată pas cu pas cum funcționează programul:

- Start → Input/Output → Process → End
- Include săgeți pentru a arăta fluxul pașilor



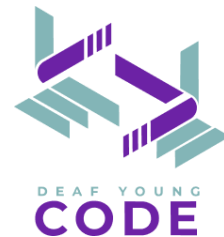
3.

Recunoașterea limbajului mimico-gestual:

Folosind biblioteci de învățare automată, Python poate ajuta la construirea unor sisteme care interpretează gesturile din limbajul semnelor în text sau vorbire.



Rezumat



- **C:** Control procedural, rapid, la nivel scăzut
- **C++:** Procedural + Orientat pe obiecte, Folosită în jocuri/aplicații
- **C#:** Modern, orientat pe obiecte, folosit în aplicații Windows/web și Unity
- Toate trei împart logica de bază a programării: variabile, bucle, funcții
- Diagramele vizuale ajută la înțelegerea fluxului programelor

Vă mulțumesc pentru atenție!



**Rămâneți pe aproape
pentru următoarea
prezentare!**

Număr proiect: 2023-2-IT03-KA220-YOU-000179130



Co-funded by
the European Union