

C, C++ and C#



DEAF YOUNG
CODE

Project number : 2023-2-IT03-KA220-YOU-000179130



Co-funded by
the European Union

Learning Objectives



By the end of this lesson, students will be able to:

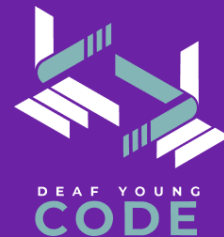
- Understand what C, C++, and C# are.
- Know the main differences between these languages.
- Read and understand simple examples in each language.
- Recognize basic programming concepts: variables, loops, and functions.
- Connect code concepts to visual logic (flowcharts, diagrams).



Video



What is C?



Definition:

- C is a high-level programming language created in the 1970s.
- Used to write fast programs and control computer hardware.

Characteristics:

- Procedural programming (step by step)
- Simple and fast
- Used in system software (e.g., operating systems)



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



What is C++?



Definition:

- C++ is an extension of C.
- Adds object-oriented programming (OOP) features: classes, objects.

Characteristics:

- Can do everything C can do
- Supports objects and classes
- Widely used in games, apps, and graphics software



Example (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**



What is C# (C Sharp)?



Definition:

- C# is a modern language created by Microsoft.
- Designed for Windows apps, web apps, and games (Unity).

Characteristics:

- Object-oriented
- Easy to use, modern syntax
- Works with .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)



Key Differences

Feature	C	C++	C#
Year	1972	1983	2000
Paradigm	Procedural	Procedural + OOP	Object-oriented
Main Use	Systems	Apps, Games	Apps, Web, Unity
Complexity	Simple	Medium	Modern, easy



Simple Comparison Example

Task: Print “Hello, World!”



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



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



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



Hands-On Activity



Activity:

1. Write a small program in any language that prints your name.
2. Draw a flowchart of what happens step by step.

1.

Task: Write a program in any of the three languages that prints your name.

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

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

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



Hands-On Activity



2. Draw a Flowchart: Show how the program works step by step:

- Start → Input/Output → Process → End
- Include arrows to show the flow of steps



3.

Sign Language Recognition:

Using machine learning libraries, Python can help build systems that interpret sign language gestures into text or speech.



Summary



- **C:** Procedural, fast, low-level control
- **C++:** Procedural + Object-oriented, used in games/apps
- **C#:** Modern, object-oriented, used in Windows/web apps and Unity
- All three share basic programming logic: variables, loops, functions
- Visual diagrams help understand program flow

Thank you for your attention!



Stay tuned for the next presentation!

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Education and Culture Executive Agency (EACEA). Neither the European Union nor EACEA can be held responsible for them.

Project number : 2023-2-IT03-KA220-YOU-000179130



Co-funded by
the European Union