COMPARISON OF PROGRAMMING CONSTRUCTS BETWEEN C++ AND C#

ROHAN JHA: 2021A7PS2721G
ATHARVA SUNDER: 2021A7PS0391G
AARYAN JAIN:2021A7PS2060G
URAV BHATT: 2021A7PS3057G

WHERE C++ FAILS:-

The compiled machine code produced by a C++ code is platform-specific. To run a C++ program on different platforms, you generally need to recompile it for each target architecture.

C++ code is typically compiled directly to machine code, which is specific to the target architecture. It results in native, unmanaged code that can be executed without the need for a runtime environment.

Memory in C++ needs to be managed directly by programmer and it can’t access libraries of other programming languages.

WHAT IS C#:-

C# is a general purpose high level language which supports multiple paradigms which encompasses strong-typing, lexically coped, imperative, declarative, generic, object-oriented and component-based programming.

It was introduced along with .NET Framework and Visual Studio, both of which are closed source. It uses a cross-platform compiler and takes the best of C and C++, to create a truely modernized language.

C# code is compiled into an intermediate language (CIL) which can then be interpreted by Common Language Runtime (CLR) so that it can be executed as native machine code on any computer.

C# DESIGN GOALS:-

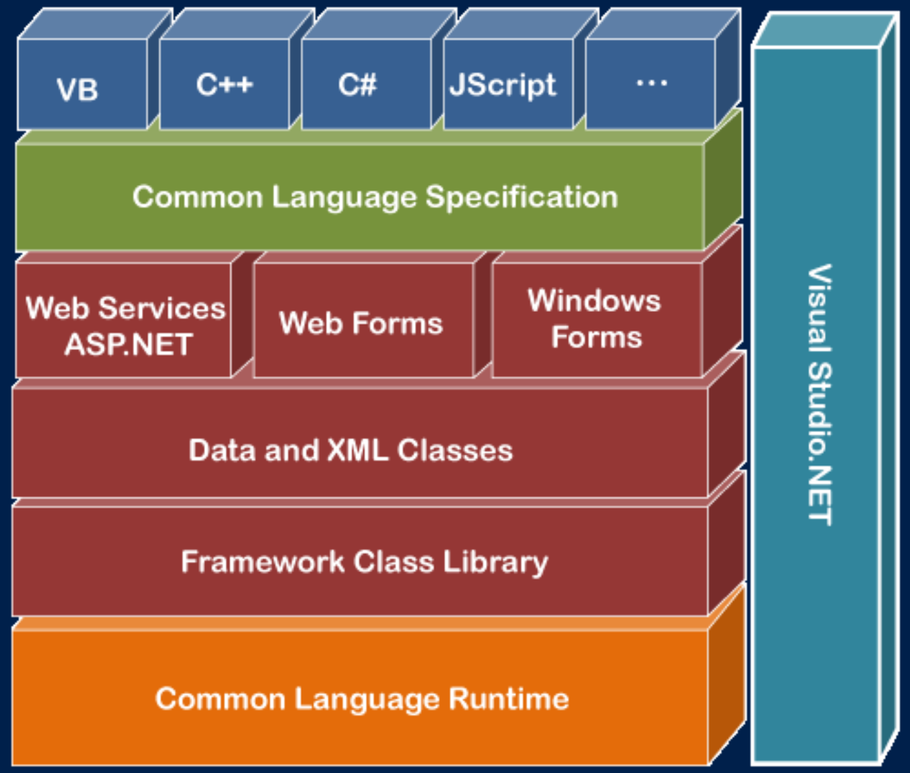
- **The language, and implementations thereof, provides support for software engineering principles such as strong type checking, array bounds checking, detection of attempts to use uninitialized variables, and automatic garbage collection. Software robustness, durability, and programmer productivity are important.**

- **Source code portability is very important, as is programmer portability, especially for those programmers already familiar with C and C++, and support for internationalization is very important.**

- **C# is intended to be suitable for writing applications for both hosted and embedded systems, ranging from the very large that use sophisticated operating systems, down to the very small having dedicated functions.**

.NET FRAMEWORK:-

The .NET framework allows users to use libraries of different programming languages along with C# making it easier to build, manage, and deploy the code.



Since CLR uses common data types, .NET allows different programming languages targeting the CLR can seamlessly work together, sharing and utilizing each other's functions and libraries.

C++ VS C#:-

- C# is Concept-Oriented Language (Describes how objects are represented and accessed. It makes references (object locations) responsible for many important functions which are difficult to model via objects) while C++ is an Object-Oriented language.

- Memory management is C++ is done by programmer manually, i.e is an object is created, it must be destroyed by the programmer. C# handles memory by using garbage collectors, which will automatically trash a created object once the object's task is complete.

C++ VS C#:-

- **Compiler in C++ does bound memory check in arrays. If programmer accesses a wrong array index, compilation error will occur. Compiler in C++ doesn’t perform bound check. Hence, illegal array index will lead to incorrect result or compilation faults.**

- **C# has a separate safe and unsafe mode. In unsafe mode, section of C# isn’t managed by Common Runtime(CLR) of .NET framework. Pointers aren’t allowed by default in C# but can be used in unsafe mode. C# behaves as C++ in unsafe mode.**

- **C++ compiles to machine code, while C# compiles to CLR(Common Language Runtime) which is interpreted by JIT in ASP.NET. C# can compile codes faster than C++.**

WHERE CAN C# BE USED :-

- **WEB DEVELOPMENT :-**

As a part of the .NET platform, C# is a natural fit for building dynamic websites and applications and integrated to work with Blazor Framework. Its object-oriented nature makes it perfect for developing websites that boast high efficiency and are easily scalable.

- **WINDOWS APPLICATIONS :-**

Since C# was developed by Microsoft, it’s only natural that it’s widely used to build Windows desktop applications. In fact, this might be the strongest use case for this language—creating applications tailored specifically to the architecture of Microsoft’s OS.

WHERE CAN C# BE USED :-

- **GAME DEVELOPMENT:-**

C# has been widely acknowledged as one of the best programming languages for gaming, especially Unity games. C# integrates with the Unity engine to provide the best environment for mobile game development—and you can even use it to develop console games with cross-platform technologies like Xamarin (Allows apps written in C# to be used on any mobile device.

There are approximately 770 million active users that use Unity game engine, the most popular game engine today, making C# responsible for more than 30% of all games being developed today.

C# LIMITATIONS :-

- **Since C# is part of the .NET ecosystem, its applications are almost exclusively for Windows-based systems. Although Microsoft is attempting to make it universal, major OS don’t support C#.**

- **Multiple inheritance through classes isn’t supported by C# while it is supported in C++.**

- **While C# is versatile and can serve quite a number of projects, that ability comes with a caveat: you need the .NET framework to make it all run smoothly. This alsso makes it a little slower than C and C++.**

- **C# is a high-level language, which doesn’t just mean that syntax and coding approaches are more abstract but also that interfacing C# products with hardware is impossible.**

THANK YOU

Fun fact: C# is known to be a C-like Object-Oriented Language and hence was about to be named ‘COOL’ at first.