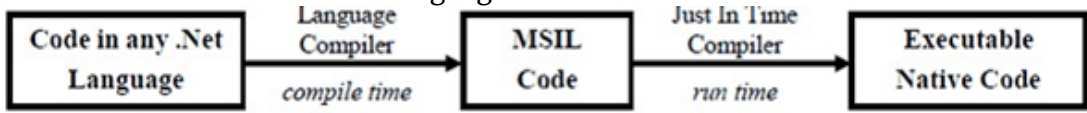


Assignment Answer of Unit-1

Prepared and Compile by Mayurkumar Marolia

1.	<p>What is .NET Framework?</p> <p>A programming infrastructure created by Microsoft for building, deploying, and running applications and services that use .NET technologies, such as desktop applications and Web services.</p>
	<p>Brief is MSIL?</p> <p>When we compile our .Net code then it is not directly converted to native/binary code; it is first converted into intermediate code known as MSIL code which is then interpreted by the CLR. MSIL is independent of hardware and the operating system. MSIL is the same for all .Net languages. MSIL is further converted into native code.</p>  <pre> graph LR A[Code in any .Net Language] -- "Language Compiler compile time" --> B[MSIL Code] B -- "Just In Time Compiler run time" --> C[Executable Native Code] </pre>
	<p>What is the purpose of JIT?</p> <p>Just - in - Time (JIT) compiler, which compiles MSIL into native code that is specific to the OS and machine architecture being targeted. Only at this point can the OS execute the application. The just - in - time part of the name reflects the fact that MSIL code is only compiled as, and when, it is needed.</p>
	<p>What is MSIL?</p> <p>It is language independent code. When you compile code that uses the .NET Framework library, you don't immediately create operating system - specific native code. Instead, you compile your code into Microsoft Intermediate Language (MSIL) code. The MSIL code is not specific to any operating system or to any language.</p>
	<p>What is the role of Garbage Collection in .NET Framework?</p> <p>.NET garbage collection works by inspecting the memory of your computer every so often and removing anything from it that is no longer needed. There is no set time frame for this; it might happen thousands of times a second, once every few seconds.</p>
	<p>What is the CLR? What is the role of CLR in .NET Framework?</p> <p>The Common Language Runtime (CLR), the virtual machine component of Microsoft's .NET framework, manages the execution of .NET programs. A process known as just-in-time compilation converts compiled code into machine instructions which the computer's CPU then executes.</p>
	<p>What is Class Library?</p> <ul style="list-style-type: none"> The Framework Class Library (FCL) is a standard library and Microsoft's .NET Framework implementation of the Standard Libraries as defined in the Common Language

Assignment Answer of Unit-1

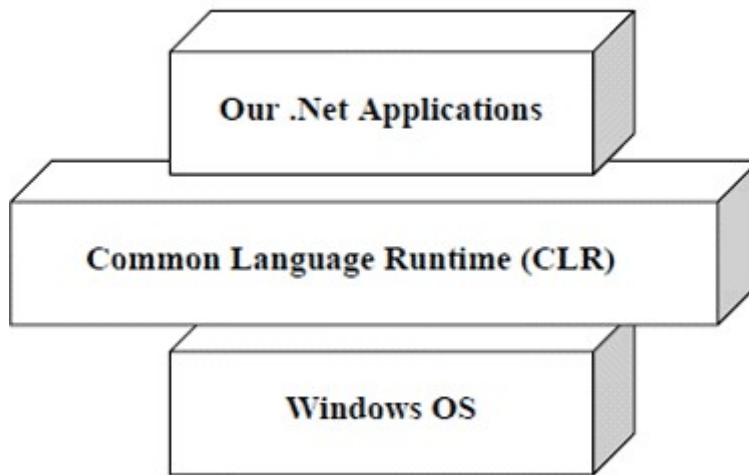
Prepared and Compile by Mayurkumar Marolia

	<p>Infrastructure.</p> <ul style="list-style-type: none"> • The FCL is a collection of reusable classes, interfaces and value types. • The Base Class Library (BCL) is the core of the FCL and provides the most fundamental functionality, which includes classes in namespaces System, System.CodeDom, System.Collections, System.Diagnostics, System.Globalization, System.IO, System.Resources and System.Text.
	<p>What do you mean by Native Code?</p> <p>The code to be executed must be converted into a language that the target operating system understands, known as native code. This conversion is called compiling code, an act that is performed by a compiler.</p> <p>Under the .NET Framework, however, this is a two - stage process. With help of MSIL and JIT.</p>
	<p>Describe the .NET Framework Features.</p> <p>The .NET Framework (pronounced <i>dot net</i>) is a proprietary, partially open source freeware software framework developed by Microsoft that runs primarily on Microsoft Windows. It includes a large class library known as Framework Class Library (FCL) and provides language interoperability (each language can use code written in other languages) across several programming languages. Programs written for .NET Framework execute in a software environment (as contrasted to hardware environment), known as Common Language Runtime (CLR), an application virtual machine that provides services such as security, memory management, and exception handling. FCL and CLR together constitute .NET Framework.</p> <ul style="list-style-type: none"> ✓ It is a platform for application developers. ✓ It is a Framework that supports Multiple Language and Cross language integration. ✓ It has IDE (Integrated Development Environment). ✓ Framework is a set of utilities or can say building blocks of your application system. ✓ .NET Framework provides GUI in a GUI manner. ✓ .NET Framework provides interoperability between languages i.e. Common Type System (CTS) . ✓ .NET Framework also includes the .NET Common Language Runtime (CLR), which is responsible for maintaining the execution of all applications developed using the .NET library. ✓ The .NET Framework consists primarily of a gigantic library of code.
	<p>Explain .NET Architecture with its components.</p>

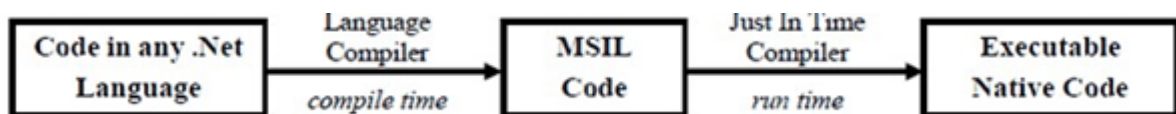
Assignment Answer of Unit-1

Prepared and Compile by Mayurkumar Marolia

1. **Common Language Runtime (CLR):** The heart of the .Net Framework. It is also called the .Net runtime. It resides above the operating system and handles all .Net applications. It handles garbage collection, Code Access Security (CAS) etc.



2. **Microsoft Intermediate Language (MSIL) Code:** When we compile our .Net code then it is not directly converted to native/binary code; it is first converted into intermediate code known as MSIL code which is then interpreted by the CLR. MSIL is independent of hardware and the operating system. Cross language relationships are possible since MSIL is the same for all .Net languages. MSIL is further converted into native code.



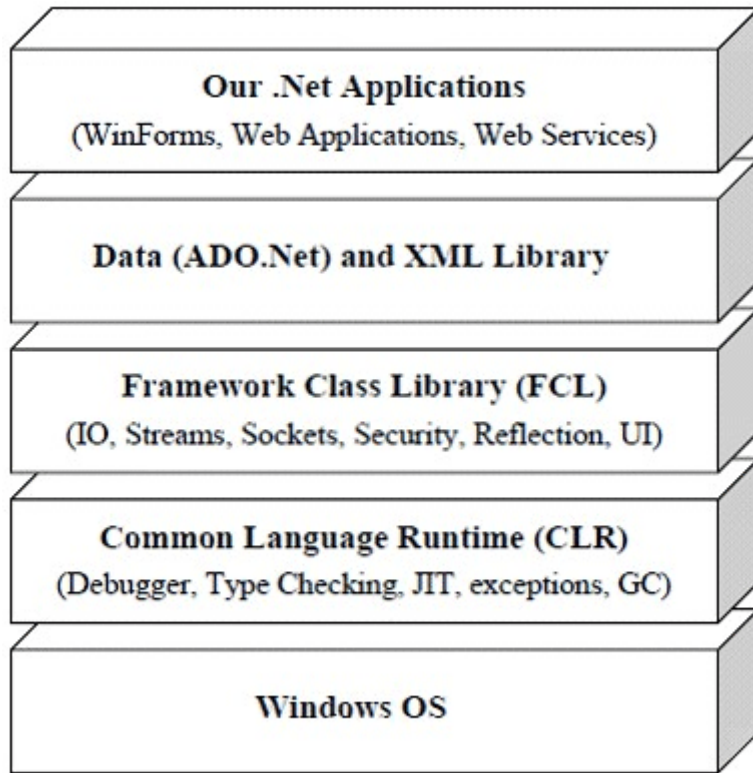
3. **Just in Time Compilers (JIT):** It compiles IL code into native executable code (exe or dlls). Once code is converted to IL then it can be called again by JIT instead of recompiling that code.
4. **Framework class library:** The .Net Framework provides a huge class library called FCL for common tasks. It contains thousands of classes to access Windows APIs and common functions like string manipulations, Data structures, stream, IO, thread, security etc.
5. **Common Language Specification (CLS):** What makes a language to be .Net compliant? Answer is CLS. Microsoft has defined some specifications that each .Net language has to follow. For e.g.: no pointer, no multiple inheritances etc.
6. **Common Type System (CTS):** CTS defines some basic data types that IL can understand. Each .Net compliant language should map its data types to these standard data types. This makes it possible for two .Net compliant languages to communicate by passing/receiving

Assignment Answer of Unit-1

Prepared and Compile by Mayurkumar Marolia

parameters to and from each other. For example CTS defines Int32 for C# int and VB integer data types.

7. **The .Net Framework:** Is a combination of CLR, FCL, ADO.Net and XML classes, Web/Window applications and Web services.



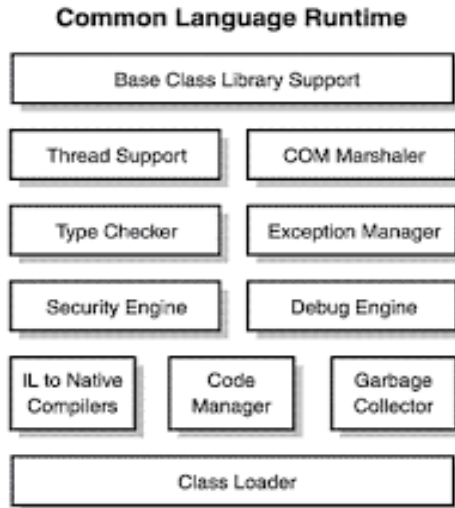
Explain the architecture of CLR.

The Common Language Runtime (CLR), the virtual machine component of Microsoft's .NET framework, manages the execution of .NET programs.

A process known as just-in-time compilation converts compiled code into machine instructions which the computer's CPU then executes.

Assignment Answer of Unit-1

Prepared and Compile by Mayurkumar Marolia



Components of the CLR

- *Class Loader*
Used to load all classes at run time.
- *MSIL to Native code*
The Just In Time (JIT) compiler will convert MSIL code into native code.
- *Code Manager*
It manages the code at run time.
- *Garbage Collector*
It manages the memory. Collect all unused objects and deallocate them to reduce memory.
- *Thread Support*
It supports multithreading of our application.
- *Exception Handler*
It handles exceptions at run time.

- **Functions of the CLR**

- Convert code into CLI.
- Exception handling
- Type safety
- Memory management (using the Garbage Collector)

Assignment Answer of Unit-1

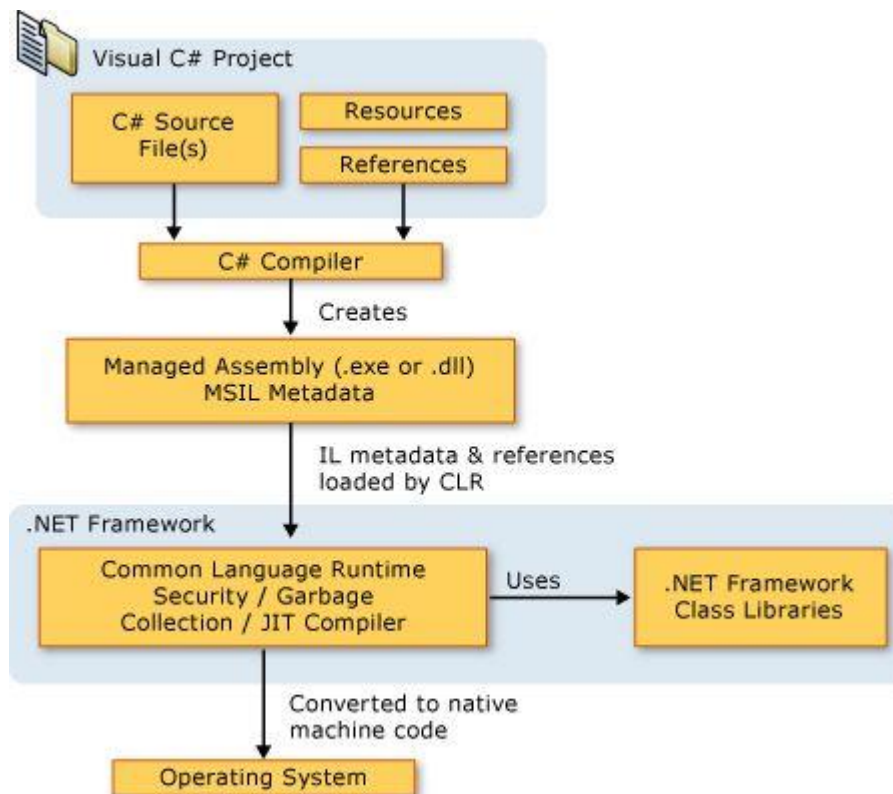
Prepared and Compile by Mayurkumar Marolia

- Security
- Improved performance
- Language independency
- Platform independency
- Architecture independency

Working of CLR

- Programmers write code in any language, including VB.Net, C# and F# when they compile their programs into an intermediate form of code called CLI in a portable execution file (PE) that can be managed and used by the CLR and then the CLR converts it into machine code to be will executed by the processor.
- The information about the environment, programming language, its version and what class libraries will be used for this code are stored in the form of metadata with the compiler that tells the CLR how to handle this code.

The CLR allows an instance of a class written in one language to call a method of the class written in another language.



Assignment Answer of Unit-1

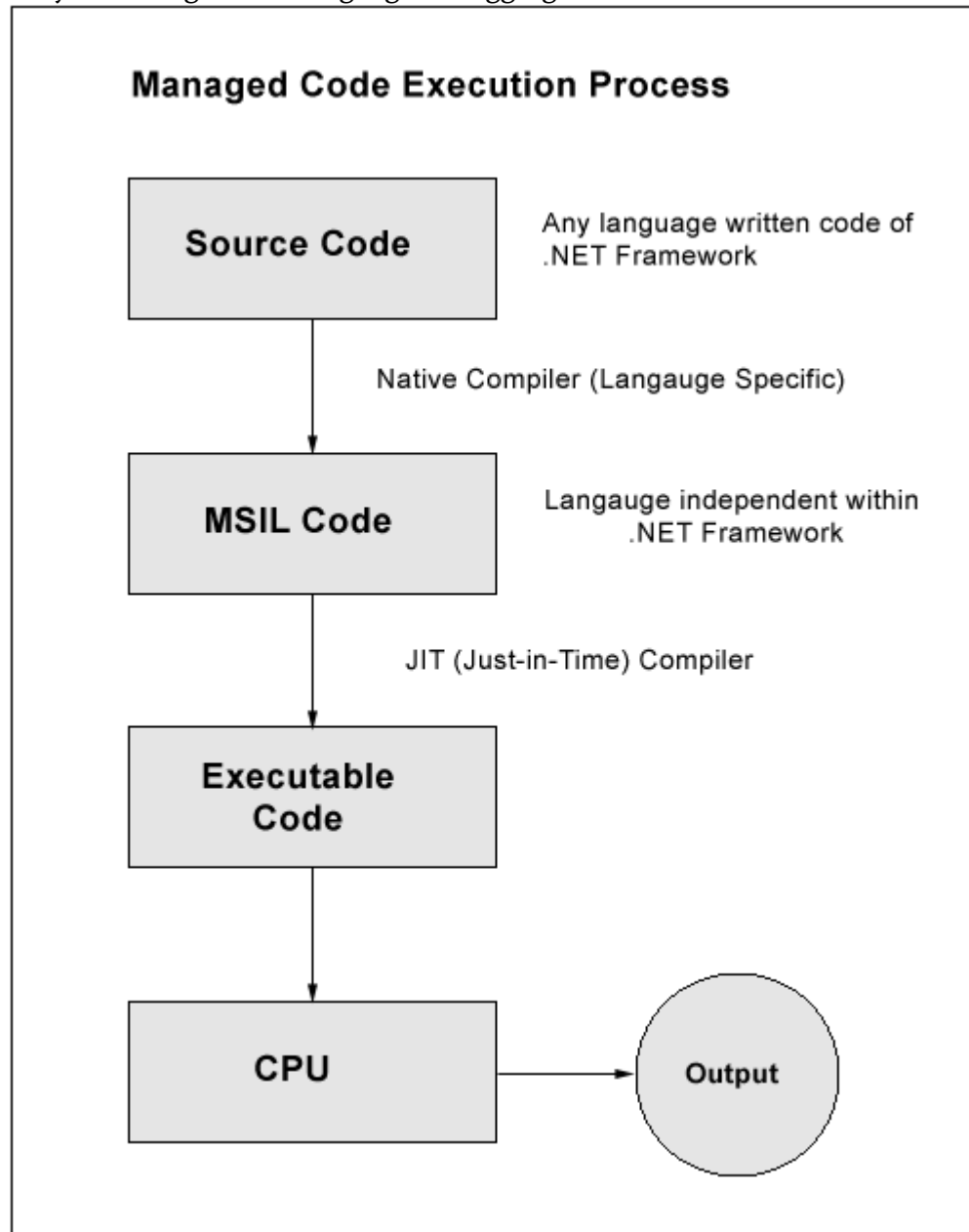
Prepared and Compile by Mayurkumar Marolia

Explain the process of Managed Code execution.

The resource, which is with in your application domain is, managed code. The resources that are within domain are faster.

The code, which is developed in .NET framework, is known as managed code. This code is directly executed by CLR with help of managed code execution. Any language that is written in .NET Framework is managed code.

Managed code uses CLR which in turns looks after your applications by managing memory, handling security, allowing cross - language debugging, and so on.



Explain the process of Unmanaged Code execution.

Assignment Answer of Unit-1

Prepared and Compile by Mayurkumar Marolia

The code, which is developed outside .NET, Framework is known as unmanaged code. Applications that do not run under the control of the CLR are said to be unmanaged, and certain languages such as C++ can be used to write such applications, which, for example, access low - level functions of the operating system. Background compatibility with code of VB, ASP and COM are examples of unmanaged code.

Unmanaged code can be unmanaged source code and unmanaged compile code.

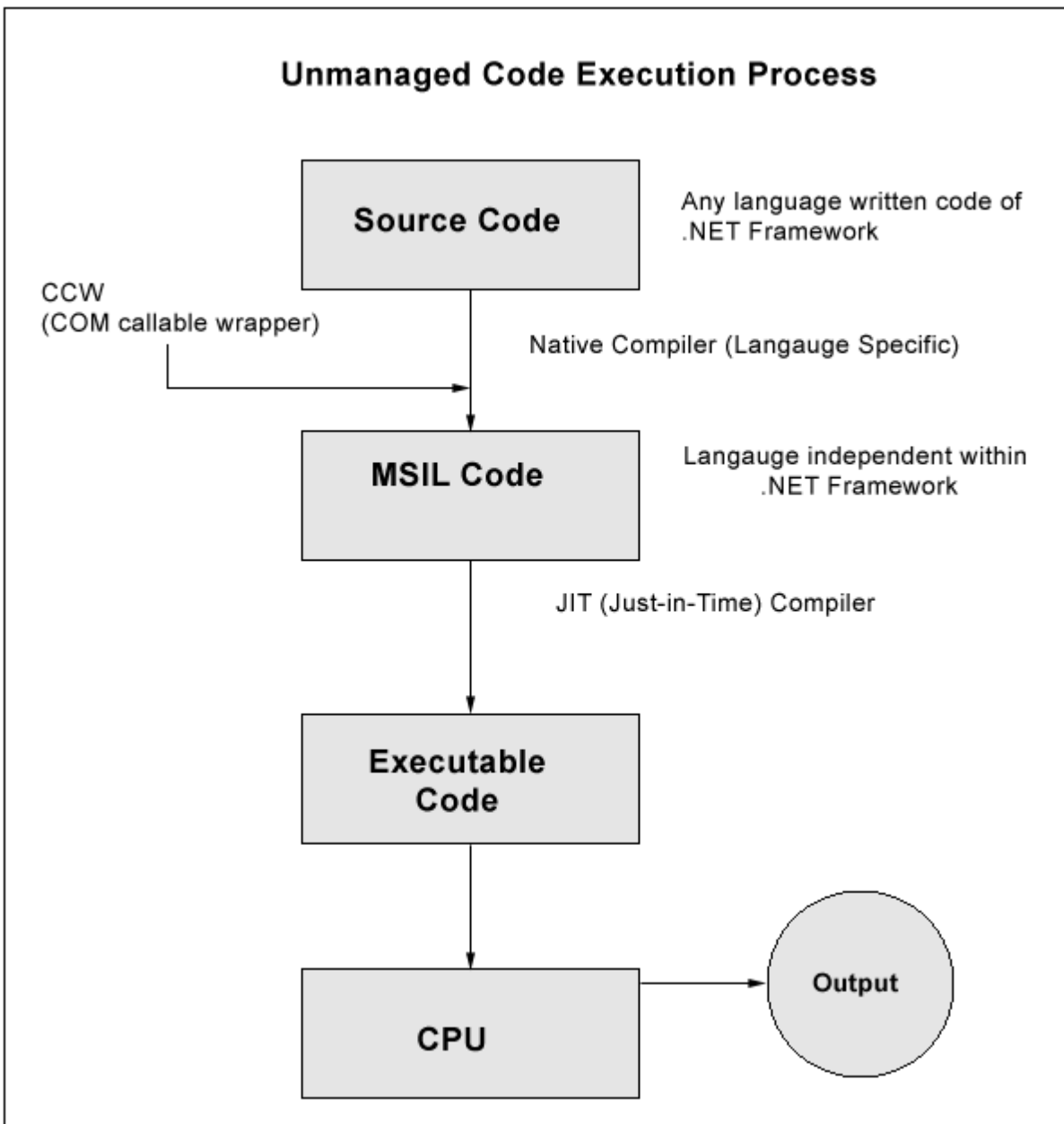
Unmanaged code is executed with help of wrapper classes.

Wrapper classes are of two types: CCW (COM callable wrapper) and RCW (Runtime Callable Wrapper).

Wrapper is used to cover difference with the help of CCW and RCW.

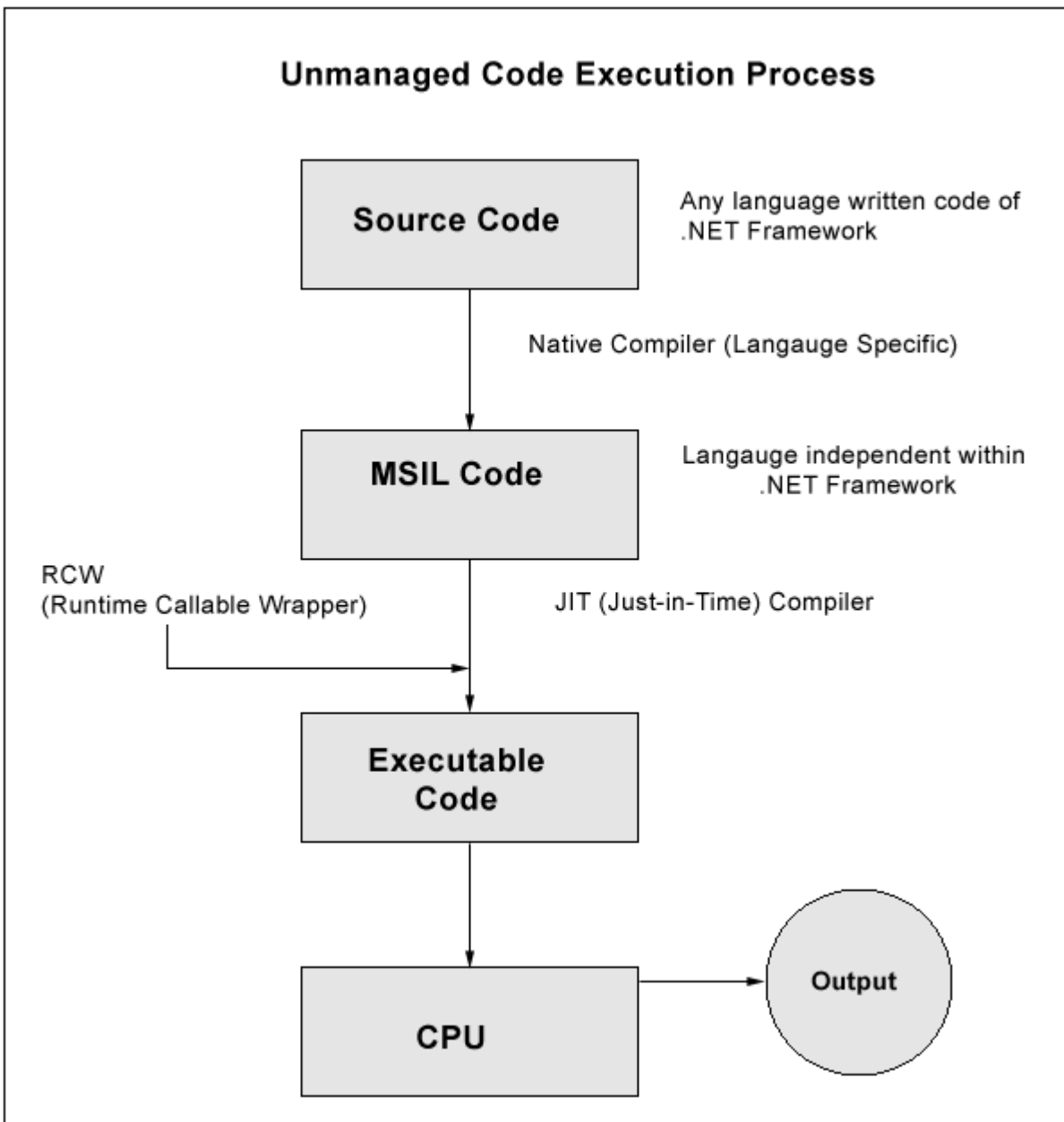
COM callable wrapper unmanaged code

Assignment Answer of Unit-1
Prepared and Compile by Mayurkumar Marolia



Runtime Callable Wrapper unmanaged code

Assignment Answer of Unit-1
Prepared and Compile by Mayurkumar Marolia



Explain Garbage Collection.

One of the most important features of managed code is the concept of garbage collection. This is the .NET method of making sure that the memory used by an application is freed up completely when the application is no longer in use.

Prior to .NET this was mostly the responsibility of programmers, and a few simple errors in code could result in large blocks of memory mysteriously disappearing as a result of being allocated to the wrong place in memory. That usually meant a progressive slowdown of your computer followed by a system crash.

Assignment Answer of Unit-1
Prepared and Compile by Mayurkumar Marolia

	<p>.NET garbage collection works by inspecting the memory of your computer every so often and removing anything from it that is no longer needed. There is no set time frame for this; it might happen thousands of times a second, once every few seconds, or whenever, but you can rest assured that it will happen.</p>