# Short Answer:

Answer the following questions with complete sentences in your own words. You are encouraged to conduct your own research online or through other methods before answering the questions. If you research online, please consult multiple sources before you write down your answers. You are expected to be able to explain your answers in detail (Provide examples for each question).

1. Explain CLR and its execution process

CLR: Common Language Runtime

-basic and virtual machine component of .NET Framework.

-runtime environment in .NET Framework

-responsible for managing the execution of .NET programs disregard the programming language. (ex: Common Language Specification garbage collector, Common Type System-value type, reference type, JIT-Just In Time Compuler)

Execution process:

1. Choose the source code language compiler -C# VB.NET
2. Compile the source code to MSIL assemblies using language compiler and generates the corresponding assembly with the MSIL inside. The assembly can be .exe or .dll file.
3. Execute the code by CLR

Diagram

Description automatically generated with medium confidence

<https://www.geeksforgeeks.org/common-language-runtime-clr-in-c-sharp/>

<https://www.c-sharpcorner.com/UploadFile/ashish_2008/clr-execution-process/>

1. What are C# variables and how to declare C# variables?

Variable: named memory location we create to store data; every variable has type to determine what values can be stored.

Declaration:

Int num; --[data type] [variable name];

Initialization:

Num = 11; [variable name]=[value];

1. What are value types? Give some examples of predefined value types in C#.

Value types: simple types, enum types, struct types, nullable value types, and tuple value types.

● Value types

○ Simple types

■ Signed integral: sbyte, short, int, long

■ Unsigned integral: byte, ushort, uint, ulong

■ floating-point: float, double

■ High-precision decimal floating-point: decimal

■ Unicode characters: char

■ Boolean: bool

○ Enum types

■ User-defined types of the form enum E {...}. An enum type is a distinct type with named constants.

Every enum type has an underlying type, which must be one of the eight integral types. The set of

values of an enum type is the same as the set of values of the underlying type.

○ Struct types

■ User-defined types of the form struct S {...}

○ Nullable value types

■ Extensions of all other value types with a null value

○ Tuple value types

■ User-defined types of the form (T1, T2, ...

The predefined reference types are object and string. The type object is the ultimate base type of all other types. The type string is used to represent Unicode string values.

1. Give examples for reference types and what is the difference between value type and reference type.

Reference types: string, class types, interface types, array types, and delegate types

C# provides the following built-in reference types:

○ string

○ dynamic

○ object

● The following keywords are used to declare reference types:

○ class

○ delegate

○ interface

○ record

A Value Type holds the data within its own memory allocation and a Reference Type contains a pointer to another memory location that holds the real data. Reference Type variables are stored in the heap while Value Type variables are stored in the stack.

Stack is used for static memory allocation and Heap for dynamic memory allocation, both stored in the computer's RAM

Boxing is converting a value type to a reference type. Unboxing is to extract the value type of an object.

Ex: int a = 23; Obj b = a;

1. What is immutable? Is string immutable or mutable?

An immutable object is defined as an object that cannot be changed after it has been created. For many use cases, such as [Data Transfer Objects](https://www.infoworld.com/article/3562271/how-to-use-data-transfer-objects-in-aspnet-core-31.html), immutability is a desirable feature.

<https://www.infoworld.com/article/3564161/how-to-use-immutability-in-csharp.html#:~:text=An%20immutable%20object%20is%20defined,immutability%20is%20a%20desirable%20feature>.

String objects are immutable: they can't be changed after they've been created.

When we assign a value to a string or modify the string, C# will create a new

String.

1. What is the string intern pool in C# and why do we need the intern pool?

String Intern Pool is a special table allocated on Large Object Heap which maintains references of all strings that are created on a program. CLR keeps track of those strings that are created in memory and uses it when it needs to create the same string again.  This ensures that new memory is not used whenever the content of the string is not different.

<https://dailydotnettips.com/the-string-intern-pool/#:~:text=String%20Intern%20Pool%20is%20a,create%20the%20same%20string%20again>.

The common language runtime(CLR) conserves string storage by maintaining

a table, called the intern pool, that contains a single reference to each

unique literal string declared or created programmatically in your program.

Consequently, an instance of a literal string with a particular value only exists

once in the system.

● For example, if you assign the same literal string to several variables, the

runtime retrieves the same reference to the literal string from the intern pool

and assigns it to each variable

1. List some ways to Concatenate Strings In C#

• string.Format();

• string.Join();

• Append() method in StringBuilder

. +

. string.Contact(str1,str2)

. string interpolation

.string formatting

1. Using + operator
2. String Interpolation
3. String.Concatenate() method
4. String.Join() method
5. String.Format() method
6. StringBuilder.Append() method

<https://learn.microsoft.com/en-us/dotnet/csharp/how-to/concatenate-multiple-strings>

<https://www.c-sharpcorner.com/article/6-effective-ways-to-concatenate-strings-in-c-sharp-and-net-core/>

1. What is meant by casting a data type? What are the 2 kinds of data type conversions available in C#?

* Typecasting is also called as type conversion
* It means converting one data type into another.
* Converting smaller data type into a larger one is also called as type promotion.
* There are two type of type conversion: implicit and explicit type conversion in C.
* Implicit type conversion operates automatically when the compatible data type is found.
* Explicit type conversion requires a type casting operator.

Implicit Casting (automatically) - converting a smaller type to a larger type size

char -> int -> long -> float -> double

Explicit Casting (manually) - converting a larger type to a smaller size type

double -> float -> long -> int -> char

1. Will the following code compile? Why？

No, cannot implicitly convert type 'type1' to 'type2'.

double myDouble = 11.11; int myInt = myDouble;

Should do typecasting:

double myDouble = 11.11;

int myInt = (int)myDouble;

1. What is the ***dynamic*** type? How does it differ from ***var***?

The dynamic type indicates that the use of the variable and references to its members

escapes compile-time type checking. Instead, it resolves the type at run time.

The dynamic types change types at run-time based on the assigned value, so the type

dynamic only exists at compile-time, not run-time.

The dynamic type variables are converted to other types implicitly.

Table

Description automatically generated

Dynamic memory allocation use heap, contains the data of reference types, relatively large access speed is slow.

1. What is the nullable value type?

You typically use a nullable value type when you need to represent the undefined value of an underlying value type. For example, a Boolean, or bool, variable can only be either true or false. However, in some applications a variable value can be undefined or missing.

Nullable value types

■ Extensions of all other value types with a null value

Starting from C# 8.0, A nullable value type represents all values of its underlying value type

and an additional null value. For example, you can assign any of the following three values to a

bool? variable: true, false, or null.

● The null-coalescing operator ?? returns the value of its left-hand operand if it isn’t null;

otherwise, it evaluates the right-hand operand and returns its result. The ?? Operator doesn’t

evaluate its right-hand operand if the left-hand operand evaluates to non-null

<https://learn.microsoft.com/en-us/dotnet/csharp/language-reference/builtin-types/nullable-value-types>

1. What are the results of the following expressions? Please include the calculation process. (logical operator)
2. 0101 and 0110 = 0100 =4
3. 0101 or 0110 = 0111 =7
4. 0101 xor 0110 = 0011 =3

|  |  |  |
| --- | --- | --- |
| 1) 5 | & | 6 |
| 2) 5 | | | 6 |
| 3) 5 | ^ | 6 |
|  |  |  |

1. Why do we need to use the break statement in the Switch statement?

The break statement is used inside the switch to terminate a statement sequence. When a break statement is reached, the switch terminates, and the flow of control jumps to the next line following the switch statement.

<https://www.geeksforgeeks.org/switch-statement-cc/>

1. What are access modifiers and their corresponding scopes in C#?

● public

accessible everywhere

● private

accessible only in current class

● protected

accessible in the same class or derived class

● internal

accessible in the same assembly

● protected internal

accessible in the same class, derived class(same assembly, same project, different assembly, different project), non-derived class(same assembly, same project)

● private protected

Accessible in the same class or derived class from the same assembly (same project)

1. Can you specify an access modifier for enumeration members?

No, Enumeration members are always public, and no access modifiers can be applied.

<https://learn.microsoft.com/en-us/dotnet/csharp/programming-guide/classes-and-structs/access-modifiers>

1. What is assembly?

An assembly is a collection of types and resources that are built to work

together and form a logical unit of functionality. Assemblies take the form of

executable (.exe) or dynamic link library (. dll) files and they are a single unit of

deployment of .NET applications.

# Coding Questions:

Write code in C# to solve the following problems. Please write your own answers. You are highly encouraged to present more than one way to answer the questions. Please follow best practices when you write the code so that it is easily readable, maintainable, and efficient. Clearly state your assumptions if you have any. You may discuss with others on the questions, but please write your own code.

1. Develop a mathematical Calculator
   1. (2 Variables -- X=5,Y=7) -->> Add, Sub, Mul, Div
   2. (3 Variables -- X=5,Y=6,Z=7) -->> Add, Sub, Mul, Div

The input contains two arrays, an array of variables and an array of operators: input1 = [5,6,14,7], input2 = [“Add”, “Sub”, “Div”]

The precedence of operator needs to be taken care of, for example, the above example should be 5 + 6 - (14 / 7) = 9 instead of (5 + 6 - 14) / 7 = 0 (if the end result is not an integer, output the floor of the decimal result)

Assumption:

1. No parentheses
2. Input will always be valid
3. input2.length = input1.length - 1
4. Write a C# program to convert minutes into a number of years and days.

Test Data

Input the number of minutes: 3456789 Expected

Output : 3456789 minutes is approximately 6 years and 210 days

1. Using only the programming techniques you learned in this lesson, write an application that calculates the squares and cubes of the numbers from 0 to 10 and prints the resulting values in a table format, as shown below. (Build-in functions are not acceptable) number square cube

. . .

10 100 1000

|  |  |  |
| --- | --- | --- |
| og | sq | cube |
| 0 | 0 | 0 |
| 1 | 1 | 1 |
| 2 | 4 | 8 |

1. (Print a table) Write a program that displays the following table:

|  |  |  |
| --- | --- | --- |
| a | b | pow(a, b) |
| 1 | 2 | 1 |
| 2 | 3 | 8 |
| 3 | 4 | 81 |
| 4 | 5 | 1024 |
| 5 | 6 | 15625 |

1. (Occurrence of max numbers) Write a program that reads integers, finds the largest of them, and counts its occurrences. Assume that the input ends with the number 0. Suppose that you entered 3 5 2 5 5 5 0; the program finds that the largest is 5 and the occurrence count for 5 is 4.

Enter numbers: 3 5 2 5 5 5 0

The largest number is 5

The occurrence count of the largest number is 4

1. Given a non-empty array of integer nums, every element appears twice except for one. Find that single one. Could you implement a solution with a linear runtime complexity without using extra memory? (Hint: xor)

Input: nums = [2,2,1] Output: 1

Input: nums = [4,1,2,1,2] Output: 4

1. PrintNumberInWord (nested-if, switch-case): Write a program called PrintNumberInWord which prints "ONE", "TWO",..., "NINE", "OTHER" if the int variable "number" is 1, 2,..., 9, or other, respectively.

Write the code in two ways:

1. a "nested-if" statement;
2. a "switch-case" statement.
3. Convert an array of char to an array of int.

Input: ['1', '2', '4', '6', '8']

Output: [1, 2, 4, 6, 8]

1. Practice on String Concatenation. Write examples for：

(1) + operator

1. string interpolation
2. string.Concat
3. String.Format
4. String.Join
5. StringBuilder.Append