C# ASSIGNMENT

1.1. What is the need for ‘Conversion of data type’ in C#

B. To store a value of one data type into a variable of another data type

C. To get desired data

D. To prevent situations of runtime error during change or conversion of data type

E. None of the mentioned

ANSWER : D.

2. What are the types of ‘Data Conversion’ in C#?

A. Implicit Conversion

B. Explicit Conversion

C. Implicit Conversion and Explicit Conversion

D. None of the mentioned

Answer: B

3. What is the subset of ‘int’ datatype?

A. long, ulong, ushort

B. long, ulong, uint

C. long, float, double

D. long, float, ushort

Answer: B.

4. CLR stands for .

A. Common Type System

B. Common Language Specification

C. Common Language Runtime

D. Java Virtual Machine

Answer: C.

5.What will be the output of the following C# code?

```
static void Main(string[] args)
{
int a, b, c, x; a = 90;
b = 15;
c = 3;
x = a - b / 3 + c * 2 - 1;
Console.WriteLine(x);
Console.ReadLine();
}
```

A. 90

B. 92

C. 89

D. 88

Answer: 90

6.What will be the output of the following C# code?

```
static void Main(string[] args)
{
int []a = { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10};
func(ref a); Console.ReadLine();
}
static void func(ref int[] x)
{
Console.Write(" numbers are : ");
for (int i = 0; i < x.Length; i++)
```

```
{
if (x[i] % 2 == 0)
{
x[i] = x[i] + 1;
Console.Write(x[i] +" ");
}
}}
```

A. numbers are : 3 5 7 9 11

B. numbers are : 2 4 6 8 10

C. numbers are : 2 3 4 5 6

D. none of the mentioned

Answer:B.

7. The correct way of incrementing the operators are:

A. d =+ 1

B. c += 1

C. b ++ 1

D. ++ a ++

Answer : B. c=c+1;

8.Which reference modifier is used to define reference variables?

A. &

B. ref

C. #

D. $

Answer:B

Basic questions.

1. What are the types of classes in C#?

a) i) Abstract class

ii) static class

iii concrete class

iv) nested class

v)partial class

2. What is a managed and unmanaged code?

A code which executed is executed by the Common Language Runtime(CLR)is known as Unmanaged code

A code which executed is directly executed by the operating system is known as Unmanaged code.

3. What are extension methods in C#?

It has allows to create and add new methods to existing class without creating anew child class. It does not require recompiling the code. An extension method should be static.It must have this keyword associate with class name.

4. What is the difference between an Array and ArrayList in C#?

Array is a collection of data from the similar data type. and it has the fixed length.

ArrayList is also a collection of data from the similar data type but it size of array will be dynamicly increases.

5.What is Boxing and Unboxing in C#?

Boxing: it is known as converting of value into object implictly is called boxing.

int z=10;

object ob=z; //boxing

UnBoxing: it is known as converting of object into value explictly is called Unboxing.

object ob=5;

int value=(int)ob; //unboxing

6. Difference between the Equality Operator (==) and Equals() Method in C#?

Equality Operator (==) : checking whether the two values are equals or not is called Equality Operator .

string s1="vamshi string s2="vamshi";

s1==s2; true ,because both refrence is same in pool .

Equals() : The Equals() method is used to compare the content in the string.

object st="vamshi";

object s=new string("vamshi");

st.equals(s);

7.What is the difference between constant and readonly in C#?

constant is initialized at compile time, read only keyword allow the variable to initialized at runtime. both values cant be modified.

8. What is the difference between String and StringBuilder in C#?

String: It is a collection of characters and mutable. string takes more processing time in terms of more iterations.

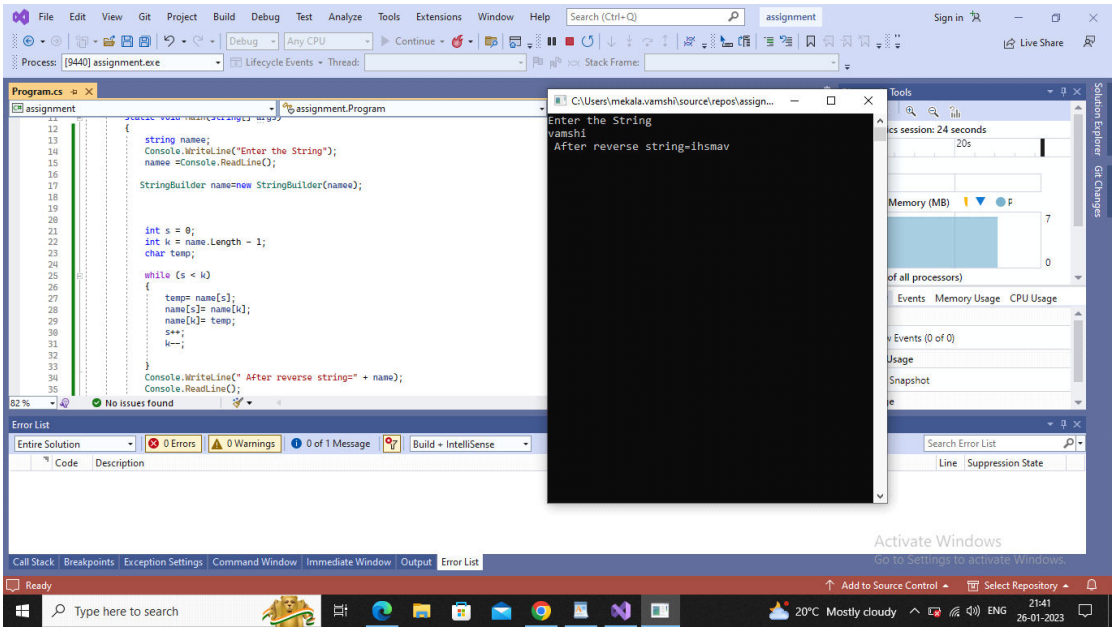
StringBuilder: It is a collection of characters and immutable. It doesnt takes more processing time in terms of more iterations.

coding problems

1. Write a program in C# Sharp to reverse a string?

```
string namee;
                Console.WriteLine("Enter the String");
                namee =Console.ReadLine();
               StringBuilder name=new StringBuilder(namee);
                int s = 0;
                int k = name.Length - 1;
                char temp;

                while (s < k)
                {
```

```
            temp= name[s];
            name[s]= name[k];
            name[k]= temp;
            s++;
            k--;
        }
        Console.WriteLine(" After reverse string=" + name);
        Console.ReadLine();
    }
```

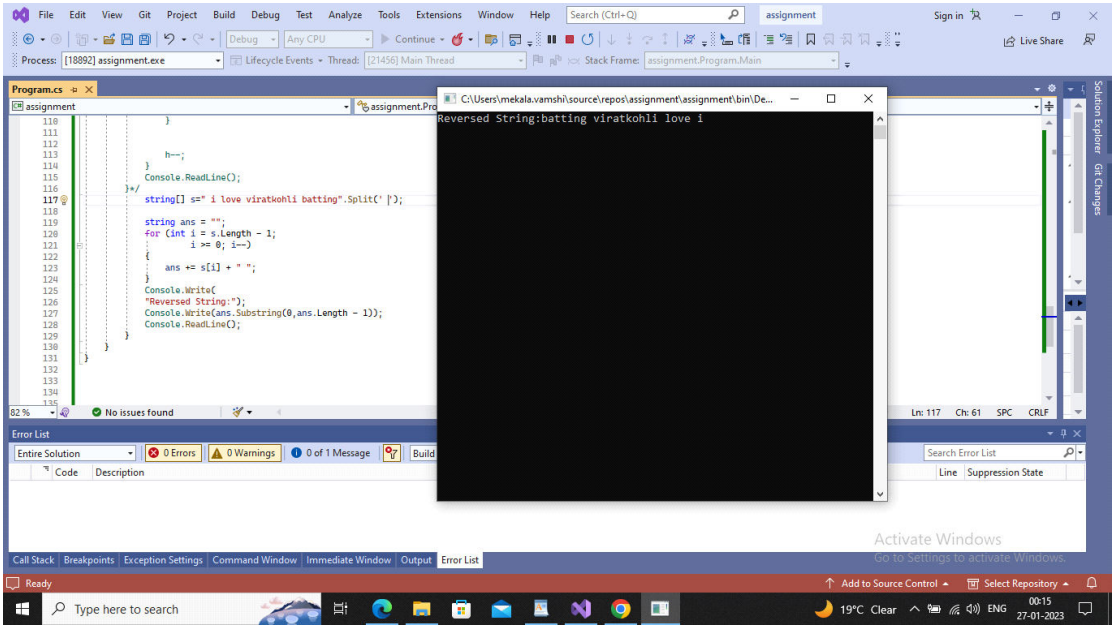


2.Write a program in C# Sharp to reverse the order of the given words?

code

```
 string[] s=" i love viratkohli batting".Split(' ');


        string ans = "";
        for (int i = s.Length - 1;
                i >= 0; i--)
        {
```

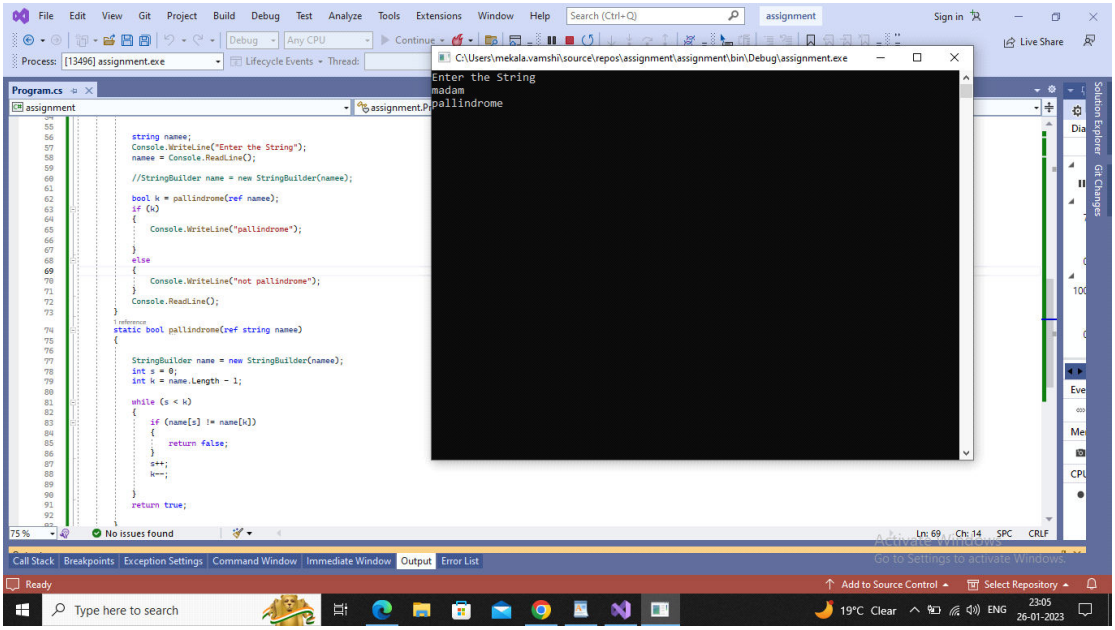
```
        ans += s[i] + " ";
    }
    Console.Write(
    "Reversed String:");
    Console.Write(ans.Substring(0,ans.Length - 1));
    Console.ReadLine();
```



3.Write a program in C# Sharp to find if a given string is palindrome or not?

```
 string namee;
        Console.WriteLine("Enter the String");
        namee = Console.ReadLine();

        //StringBuilder name = new StringBuilder(namee);

        bool k = pallindrome(ref namee);
        if (k)
        {
            Console.WriteLine("pallindrome");

        }
```

```
            else
            {
                Console.WriteLine("not pallindrome");
            }
            Console.ReadLine();
        }
static bool pallindrome(ref string namee)
        {

            StringBuilder name = new StringBuilder(namee);
            int s = 0;
            int k = name.Length - 1;

            while (s < k)
            {
                if (name[s] != name[k])
                {
                    return false;
                }
                s++;
                k--;

            }
            return true;
```



4. Write a C# program to find the substring from a given string.

```
string name, sub;
        int h = 2;
        while (h > 0) {
        Console.WriteLine("enter the string");
        name = Console.ReadLine();
        Console.WriteLine("enter the substring");
        sub = Console.ReadLine();
        bool k = name.Contains(sub);

          if (k)
          {
              Console.WriteLine("the substring is present");
          }
          else
          {
              Console.WriteLine("no substring");
          }
```

```
            h--;
        }
        Console.ReadLine();
```