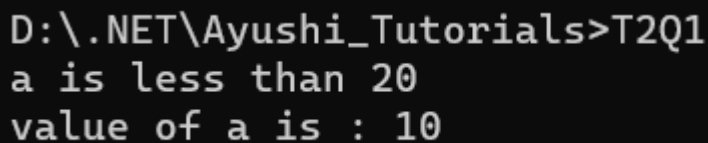# Lab Tutorial – 2

1 : Predict and write output for the following code.

```
using System;
namespace DecisionMaking
{
    class Program
    {
        static void Main(string[] args)
        {
            /* local variable definition */
            int a = 10;
            /* check the boolean condition using if statement */
            if (a < 20)
            {
                /* if condition is true then print the following */
                Console.WriteLine("a is less than 20");
            }
            Console.WriteLine("value of a is : {0}", a);
            Console.ReadLine();
        }
    }
}
```

```
D:\.NET\Ayushi_Tutorials>T2Q1
a is less than 20
value of a is : 10
```

2 : Write missing statement to get the desired output.

```
using System;
namespace DecisionMaking
{
    class T2Q2
    {
        static void Main(string[] args)
        {
```

```
        /* local variable definition */
        int a = 100;
        /* check the boolean condition */
        if (a < 20)
        {
            /* if condition is true then print the following */
            Console.WriteLine("a is less than 20");
        }
        else
        {
            /* if condition is false then print the following */

            //…………………………………Missing statement-1……………………………..//
        }
         //………………………………Missing statement-2………………………………..//
        Console.ReadLine();
    }
  }
}
```
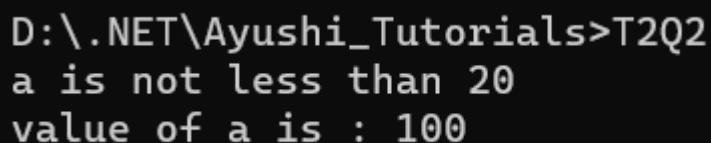
Output

a is not less than 20
value of a is : 100

```
else
{
    /* if condition is false then print the following */

    //……………………………………Missing statement-1………………………………..//
    Console.WriteLine("a is not less than 20");
}
//……………………………………Missing statement-2………………………………..//
Console.WriteLine("value of a is : {0}", a);
Console.ReadLine();
```

```
D:\.NET\Ayushi_Tutorials>T2Q2
a is not less than 20
value of a is : 100
```

3 : Correct the following code and write output for the corrected code.

```
using System;
namespace ConsoleApplication1
{
    class T2Q3
    {
        static void Main(string[] args)
        {
            char firstName = "John";
            char lastName = "Doe";
            Console.WriteLine("Name: " + firstName + " " + lastName);
            Console.WriteLine("Please enter a new first name:")
            firstName = Console.ReadLine();
            Console.WriteLine("New name: "  firstName  " "  lastName);
            Console.ReadLine();
        }
    }
}
```

```
        static void Main(string[] args)
        {
            string firstName = "John";              // char -> string
            string lastName = "Doe";                // char -> string
            Console.WriteLine("Name: " + firstName + " " + lastName);
            Console.WriteLine("Please enter a new first name:");                                    // ;
            firstName = Console.ReadLine();
            Console.WriteLine("New name: "  + firstName  + " " + lastName);          // proper use of +
            Console.ReadLine();
        }
```

```
D:\.NET\Ayushi_Tutorials>T2Q3
Name: John Doe
Please enter a new first name:
abc
New name: abc Doe
```

4 : Input two number A and B. perform different operations using different operators and different data types available in C#. (Note : Follow all the operators and data types to do above task. Use Online help whenever necessary.)

```csharp
using System;

namespace Ayushi_Tutorials
{
    internal class T2Q4
    {
        public static void Main(string[] args)
        {
            // Input two numbers
            Console.WriteLine("Enter first number:");
            int n1 = Convert.ToInt32(Console.ReadLine());
            Console.WriteLine("Enter second number:");
            int n2 = Convert.ToInt32(Console.ReadLine());

            // Arithmetic operators
            Console.WriteLine("~~~ Arithmetic Operators ~~~");
            Console.WriteLine("Addition : " + (n1 + n2));
            Console.WriteLine("Subtraction : " + (n1 - n2));
            Console.WriteLine("Multiplication : " + (n1 * n2));
            Console.WriteLine("Division : " + ((double)n1/n2));  // using double for division
            Console.WriteLine("Modulo : "+ (n1 % n2));

            // Relational Operators
            Console.WriteLine("~~~ Relational Operators ~~~");
            Console.WriteLine("Equal to : " + (n1 == n2));
            Console.WriteLine("Not equal to : " + (n1 != n2));
            Console.WriteLine("Greater than : " + (n1 > n2));
            Console.WriteLine("Less than : " + (n1 < n2));
            Console.WriteLine("Greater than or equal to : " + (n1 >= n2));
            Console.WriteLine("Less than or equal to: " + (n1 <= n2));

            // Logical Operators
            Console.WriteLine("~~~ Logical Operators ~~~");
            bool cond1 = (n1 > 0);
            bool cond2 = (n2 > 0);
            Console.WriteLine("Logical AND : " + (cond1 && cond2));
            Console.WriteLine("Logical OR : " + (cond1 || cond2));
```

```csharp
            Console.WriteLine("Logical NOT : " + (!cond1));

            // Bitwise Operators
            Console.WriteLine("~~~ Bitwise Operators ~~~");
            Console.WriteLine("Bitwise AND : " + (n1 & n2));
            Console.WriteLine("Bitwise OR : " + (n1 | n2));
            Console.WriteLine("Bitwise XOR : " + (n1 ^ n2));
            Console.WriteLine("Bitwise Complement(NOT) : " + (~n1));
            Console.WriteLine("Left shift : " + (n1 << 1));
            Console.WriteLine("Right shift : " + (n2 >> 1));

            // Assignment Operators
            Console.WriteLine("~~~ Assignment Operators ~~~");
            int x = n1;
            x = n2; Console.WriteLine("Simple assignment : " + x);
            x += n2; Console.WriteLine("Add and assign : " + x);
            x -= n2; Console.WriteLine("Subtract and assign : " + x);
            x *= n2; Console.WriteLine("Multiply and assign : " + x);
            x /= n2; Console.WriteLine("Divide and assign : " + x);
            x %= n2; Console.WriteLine("Modulus and assign : " + x);

            // Unary Operators
            Console.WriteLine("~~~ Unary Operators ~~~");
            int y = n1;
            Console.WriteLine("Unary plus : " + (+y));
            Console.WriteLine("Unary minus : " + (-y));
            Console.WriteLine("Pre Increment : " + (++y));
            Console.WriteLine("Pre Decrement : " + (--y));
            Console.WriteLine("Post Increment : " + (y++));
            Console.WriteLine("Post Decrement : " + (y--));

            // Other Operators
            Console.WriteLine("~~~ Other Operators ~~~");
            Console.WriteLine("Ternary Operator : " + (n1 > n2 ? n1 : n2));
            Console.WriteLine("Type Cast Operator : " + ((double)n1));
            Console.WriteLine($"String Interpolation : n1 = {n1}, n2 = {n2}");
        }
    }
}
```

```
Enter first number:
14
Enter second number:
25
~~~ Arithmetic Operators ~~~
Addition : 39
Subtraction : -11
Multiplication : 350
Division : 0.56
Modulo : 14
~~~ Relational Operators ~~~
Equal to : False
Not equal to : True
Greater than : False
Less than : True
Greater than or equal to : False
Less than or equal to: True
~~~ Logical Operators ~~~
Logical AND : True
Logical OR : True
Logical NOT : False
~~~ Bitwise Operators ~~~
Bitwise AND : 8
Bitwise OR : 31
Bitwise XOR : 23
Bitwise Complement(NOT) : -15
Left shift : 28
Right shift : 12
~~~ Assignment Operators ~~~
Simple assignment : 25
Add and assign : 50
Subtract and assign : 25
Multiply and assign : 625
Divide and assign : 25
Modulus and assign : 0
~~~ Unary Operators ~~~
Unary plus : 14
Unary minus : -14
Pre Increment : 15
Pre Decrement : 14
Post Increment : 14
Post Decrement : 15
~~~ Other Operators ~~~
Ternary Operator : 25
Type Cast Operator : 14
String Interpolation : n1 = 14, n2 = 25
```

5 : Rearrange the given code to correct the program. The resultant program will be to enter 5 elements into an array and print sum of these elements.

```csharp
using System;
namespace ConsoleApplication1
{
    class T2Q5
    {
        static void Main(string[] args)
        {
            for (int i = 0; i < 5; i++)
            {
                string str = Console.ReadLine();
            }
            for (int i = 0; i < 5; i++)
            {
                sum = sum + arr[i];
            }
            Console.WriteLine("Sum of Elements : {0}",sum);
            int[] arr = new int[5];
            int sum = 0;
            arr[i] = Convert.ToInt32(str);
            Console.Write("Enter Element {0}: ", i);
            Console.Read();
        }
    }
}
```
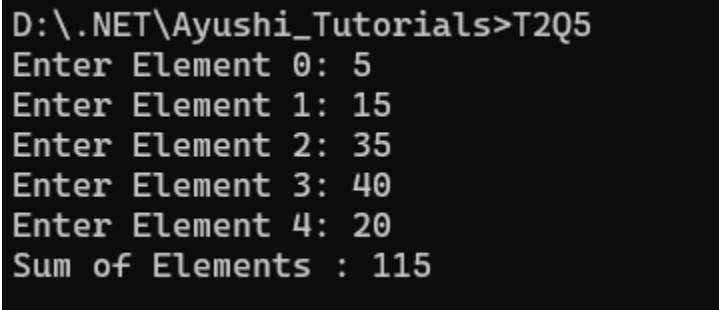
Output:

Enter Element 0: 1
Enter Element 1: 2
Enter Element 2: 3
Enter Element 3: 4
Enter Element 4: 5
Sum of Elements : 15

```csharp
static void Main(string[] args)
{
    int[] arr = new int[5];
    int sum = 0;
```

```
        for (int i = 0; i < 5; i++)
        {
            Console.Write("Enter Element {0}: ", i);
            string str = Console.ReadLine();
            arr[i] = Convert.ToInt32(str);
        }
        for (int i = 0; i < 5; i++)
        {
            sum = sum + arr[i];
        }
        Console.WriteLine("Sum of Elements : {0}", sum);
        Console.Read();
    }
```

```
D:\.NET\Ayushi_Tutorials>T2Q5
Enter Element 0: 5
Enter Element 1: 15
Enter Element 2: 35
Enter Element 3: 40
Enter Element 4: 20
Sum of Elements : 115
```

6: Write missing statement to get the desired output.

```
using System;
public class T2Q6
{
    public static void Main(string[] args)
    {
        Console.WriteLine("Hello, World!");
        Console.WriteLine("You entered the following {0} command line arguments:",
            args.Length );
        //………………………………Missing statement-1……………………………..//

        //………………………………Missing statement-2……………………………..//

        //………………………………Missing statement-3……………………………..//

        //………………………………Missing statement-4……………………………..//
    }
}
```
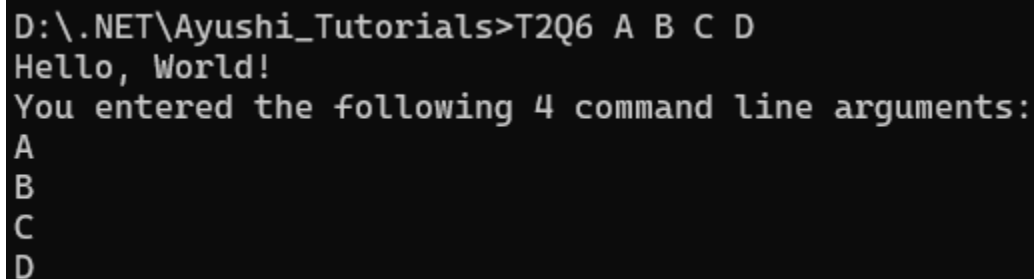
Output:

Hello, World!
You entered the following 4 command line arguments:
A
B
C
D
//………………………………Missing statement-1………………………………//
Console.WriteLine(args[0]);

//………………………………Missing statement-2………………………………//
Console.WriteLine(args[1]);

//………………………………Missing statement-3………………………………//
Console.WriteLine(args[2]);

//………………………………Missing statement-4………………………………//
Console.WriteLine(args[3]);
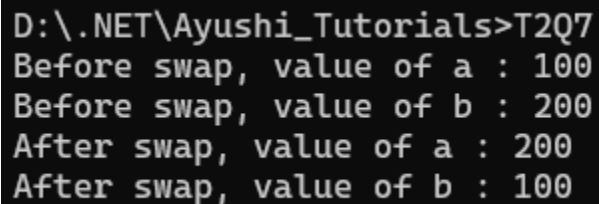
```
D:\.NET\Ayushi_Tutorials>T2Q6 A B C D
Hello, World!
You entered the following 4 command line arguments:
A
B
C
D
```

7 : Predict and write the output of the given code.

```
using System;
namespace CalculatorApplication
{
  class NumberManipulator
  {
    public void swap(ref int x, ref int y)
    {
      int temp;
      temp = x; /* save the value of x */
```

```csharp
      x = y;   /* put y into x */
      y = temp; /* put temp into y */
    }
  }
  class T2Q7
  {
    static void Main(string[] args)
    {
      NumberManipulator n = new NumberManipulator();
      /* local variable definition */
      int a = 100;
      int b = 200;
      Console.WriteLine("Before swap, value of a : {0}", a);
      Console.WriteLine("Before swap, value of b : {0}", b);
      /* calling a function to swap the values */
      n.swap(ref a, ref b);
      Console.WriteLine("After swap, value of a : {0}", a);
      Console.WriteLine("After swap, value of b : {0}", b);
      Console.ReadLine();
    }
  }
}
```

```
D:\.NET\Ayushi_Tutorials>T2Q7
Before swap, value of a : 100
Before swap, value of b : 200
After swap, value of a : 200
After swap, value of b : 100
```

```csharp
using System;
namespace CalculatorApplication
{
  class NumberManipulator
  {
    public int getValues(out int x, out int y, out int z )
    {
      Console.WriteLine("Enter the first value: ");
      x = Convert.ToInt32(Console.ReadLine());
      Console.WriteLine("Enter the second value: ");
```
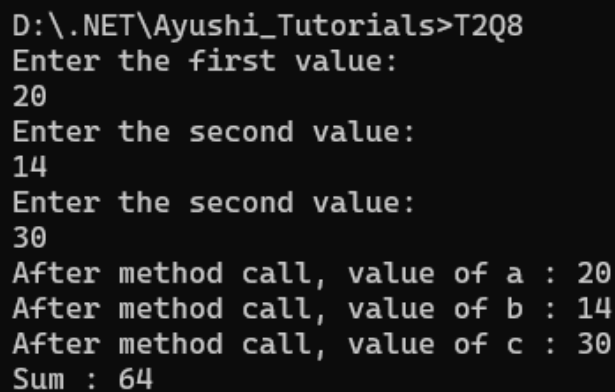
```csharp
            y = Convert.ToInt32(Console.ReadLine());
            sum = "x" + "y" + "z";
            return "sum";
        }
    }
    class T2Q8
    {
        static void Main(string[] args)
        {
            NumberManipulator n = new NumberManipulator();
            /* local variable definition */
            int a , b, c, sum;
            /* calling a function to get the values */
            sum = n.getValues(out a, out b, out c);
            Console.WriteLine("After method call, value of a : {0}", a);
            Console.WriteLine("After method call, value of b : {0}", b);
            Console.WriteLine("After method call, value of c : {0}", c);
            Console.WriteLine("Sum : {0}", );
        }
    }
}

using System;
namespace CalculatorApplication
{
    class NumberManipulator1
    {
        public int getValues(out int x, out int y, out int z)
        {
            Console.WriteLine("Enter the first value: ");
            x = Convert.ToInt32(Console.ReadLine());
            Console.WriteLine("Enter the second value: ");
            y = Convert.ToInt32(Console.ReadLine());
            Console.WriteLine("Enter the second value: ");
            z = Convert.ToInt32(Console.ReadLine());
            int sum = x + y + z;
            return sum;
        }
    }
    class T2Q8
    {
```

```
        static void Main(string[] args)
        {
            NumberManipulator1 n = new NumberManipulator1();
            /* local variable definition */
            int a, b, c, sum;
            /* calling a function to get the values */
            sum = n.getValues(out a, out b, out c);
            Console.WriteLine("After method call, value of a : {0}", a);
            Console.WriteLine("After method call, value of b : {0}", b);
            Console.WriteLine("After method call, value of c : {0}", c);
            Console.WriteLine("Sum : {0}", sum);
        }
    }
}
```

```
D:\.NET\Ayushi_Tutorials>T2Q8
Enter the first value:
20
Enter the second value:
14
Enter the second value:
30
After method call, value of a : 20
After method call, value of b : 14
After method call, value of c : 30
Sum : 64
```

9 : Given an array A containing 2*N+2 positive numbers, out of which 2*N numbers exist in pairs whereas the other two number occur exactly once and are distinct. Find the other two numbers.

Example 1 :
Input:
N = 2
arr[] = {1, 2, 3, 2, 1, 4}

Output:
3 4
Explanation:
3 and 4 occur exactly once.

Example 2 :

Input:
N = 1
arr[] = {2, 1, 3, 2}

Output:
1 3
Explanation:
1 3 occur exactly once.

```csharp
using System;
namespace Ayushi_Tutorials
{
    class T2Q9
    {
        static void Main(string[] args)
        {
            int[] arr = { 1, 2, 3, 2, 1, 4 }; // Example input
            int n = arr.Length;

            int xorAll = 0;

            // Step 1: XOR all elements
            for (int i = 0; i < n; i++)
            {
                xorAll ^= arr[i];
            }

            // Step 2: Find rightmost set bit
            int setBit = xorAll & ~(xorAll - 1);

            int x = 0, y = 0;

            // Step 3: Divide into two groups and XOR separately
            for (int i = 0; i < n; i++)
            {
                if ((arr[i] & setBit) != 0)
                    x ^= arr[i];
                else
                    y ^= arr[i];
            }
```
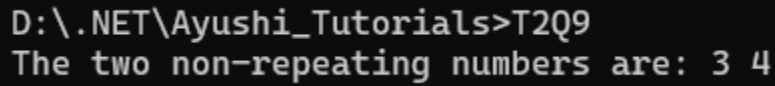
```
        Console.WriteLine("The two non-repeating numbers are: {0} {1}", x, y);
    }
  }
}
```

```
D:\.NET\Ayushi_Tutorials>T2Q9
The two non-repeating numbers are: 3 4
```

10: Given a matrix mat[][] of size N x M, where every row and column is sorted in increasing order, and a number X is given. The task is to find whether element X is present in the matrix or not.

Example 1:
Input:
N = 3, M = 3
mat[][] = 3 30 38

44 52 54

57 60 69
X = 62

Output:
0
Explanation:
62 is not present in the
matrix, so output is 0

Example 2:
Input:
N = 1, M = 6
mat[][] = 18 21 27 38 55 67
X = 55

Output:
1
Explanation:
55 is present in the
matrix at 5th cell.

Your Task:

You don't need to read input or print anything. You just have to complete the function matSearch() which takes a 2D matrix mat[][], its dimensions N and M and integer X as inputs and returns 1 if the element X is present in the matrix and 0 otherwise.

Expected Time Complexity: O(N+M).
Expected Auxiliary Space: O(1).

Constraints:

1 <= N, M <= 1005

1 <= mat[][] <= 10000000

1<= X <= 10000000

```
using System;

public class T2Q10
{
    public int matSearch(int[,] mat, int N, int M, int X)
    {
        int i = 0, j = M - 1; // start from top-right corner

        while (i < N && j >= 0)
        {
            if (mat[i, j] == X)
                return 1; // found
            else if (mat[i, j] > X)
                j--; // move left
            else
                i++; // move down
        }

        return 0; // not found
    }
    static void Main(string[] args)
    {
        T2Q10 sol = new T2Q10();
```
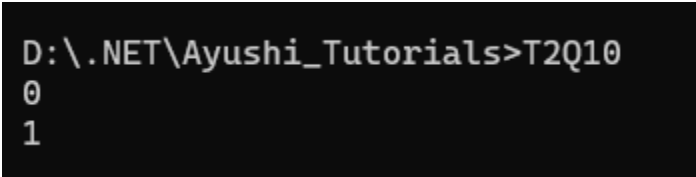
```
        // Example 1
        int[,] mat1 = { { 3, 30, 38 }, { 44, 52, 54 }, { 57, 60, 69 } };
        int N1 = 3, M1 = 3, X1 = 62;
        int res1 = sol.matSearch(mat1, N1, M1, X1);
        Console.WriteLine(res1); // Output: 0

        // Example 2
        int[,] mat2 = { { 18, 21, 27, 38, 55, 67 } };
        int N2 = 1, M2 = 6, X2 = 55;
        int res2 = sol.matSearch(mat2, N2, M2, X2);
        Console.WriteLine(res2); // Output: 1

        Console.ReadLine();
    }
}
```

```
D:\.NET\Ayushi_Tutorials>T2Q10
0
1
```

11: Write a program to find the sum of N elements of an Array.

```
using System;

namespace Ayushi_Tutorials
{
    internal class T2Q11
    {
        static void Main(string[] args)
        {
            Console.Write("Enter the number of elements (N): ");
            int N = Convert.ToInt32(Console.ReadLine());

            int[] arr = new int[N];
            int sum = 0;

            // Input elements
            for (int i = 0; i < N; i++)
            {
                Console.Write("Enter element {0}: ", i);
```
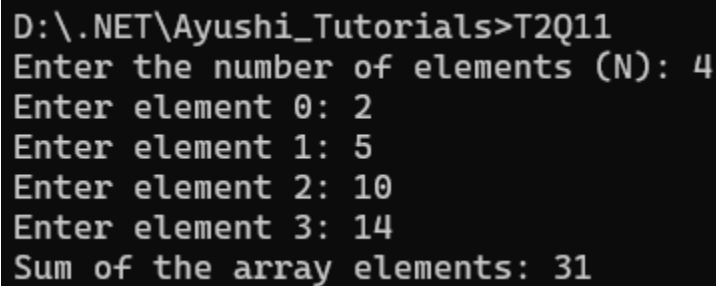
```
            arr[i] = Convert.ToInt32(Console.ReadLine());
            sum += arr[i]; // add element to sum
        }

        // Print sum
        Console.WriteLine("Sum of the array elements: {0}", sum);
    }
  }
}
```

```
D:\.NET\Ayushi_Tutorials>T2Q11
Enter the number of elements (N): 4
Enter element 0: 2
Enter element 1: 5
Enter element 2: 10
Enter element 3: 14
Sum of the array elements: 31
```

12: Write a program to find the element from an Array and print 1 if element is found else print 0.

```csharp
using System;
namespace Ayushi_Tutorials
{
  internal class T2Q12
  {
    static void Main(string[] args)
    {
      Console.Write("Enter the number of elements (N): ");
      int N = Convert.ToInt32(Console.ReadLine());

      int[] arr = new int[N];

      // Input array elements
      for (int i = 0; i < N; i++)
      {
        Console.Write("Enter element {0}: ", i);
        arr[i] = Convert.ToInt32(Console.ReadLine());
      }

      // Input element to search
      Console.Write("Enter element to search: ");
```
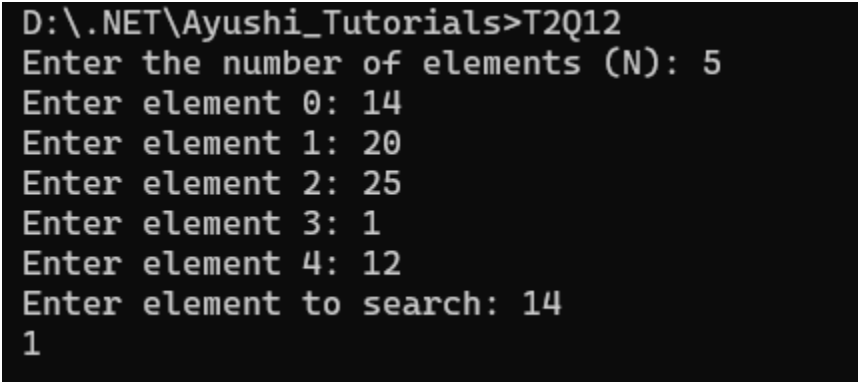
```
        int key = Convert.ToInt32(Console.ReadLine());

        int found = 0; // 0 = not found, 1 = found

        // Search loop
        for (int i = 0; i < N; i++)
        {
            if (arr[i] == key)
            {
                found = 1;
                break;
            }
        }

        Console.WriteLine(found);
        Console.ReadLine();
    }
  }
}
```

```
D:\.NET\Ayushi_Tutorials>T2Q12
Enter the number of elements (N): 5
Enter element 0: 14
Enter element 1: 20
Enter element 2: 25
Enter element 3: 1
Enter element 4: 12
Enter element to search: 14
1
```

13. Write a Program that will accept the amount and find how many minimum no of notes you required for that.
(Using the rupee notes of 1, 2, 5, 10, 20, 50, 100, 200, 500, 2000)
Input: 5748
Output:
Notes of Rs.2000 = 2 Notes of Rs.500 = 3 Notes of Rs.200 = 1
Notes of Rs.20 = 2 Notes of Rs.10 = 0 Notes of Rs.5 = 1
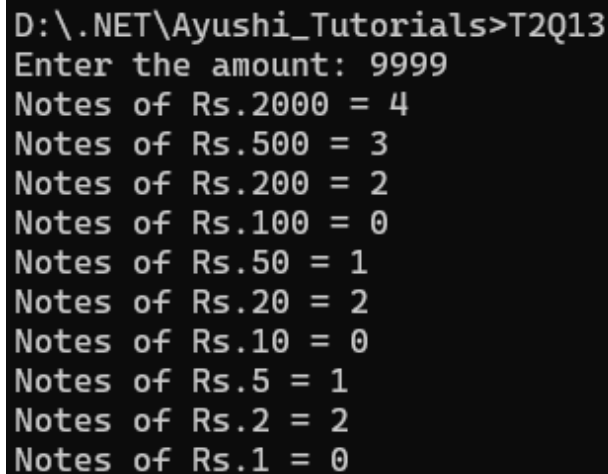**Notes of Rs.2 = 1 Notes of Rs.1 = 1**

using System;

```
namespace Ayushi_Tutorials
{
    internal class T2Q13
    {
        static void Main(string[] args)
        {
            Console.Write("Enter the amount: ");
            int amount = Convert.ToInt32(Console.ReadLine());

            int[] notes = { 2000, 500, 200, 100, 50, 20, 10, 5, 2, 1 };
            int[] count = new int[notes.Length];
            int remaining = amount;
            for (int i = 0; i < notes.Length; i++)
            {
                count[i] = remaining / notes[i]; // number of notes of this denomination
                remaining = remaining % notes[i]; // remaining amount
            }
            // Print result
            for (int i = 0; i < notes.Length; i++)
            {
                Console.WriteLine("Notes of Rs.{0} = {1}", notes[i], count[i]);
            }
            Console.ReadLine();
        }
    }
}
```

```
D:\.NET\Ayushi_Tutorials>T2Q13
Enter the amount: 9999
Notes of Rs.2000 = 4
Notes of Rs.500 = 3
Notes of Rs.200 = 2
Notes of Rs.100 = 0
Notes of Rs.50 = 1
Notes of Rs.20 = 2
Notes of Rs.10 = 0
Notes of Rs.5 = 1
Notes of Rs.2 = 2
Notes of Rs.1 = 0
```

14. Write a Program to find the eligibility of admission for a professional course based on the following criteria:

Marks in Maths >=65
Marks in Phy >=55
Marks in Chem>=50 and
Total in all three subject >=180 or
Total in Math and Physics >=140
INPUT:
Input the marks obtained in Maths :72
Input the marks obtained in Physics :65
Input the marks obtained in Chemistry :51
OUTPUT:
The candidate is eligible for admission.

```csharp
using System;

namespace Ayushi_Tutorials
{
    internal class T2Q14
    {
        static void Main(string[] args)
        {
            // Input marks
            Console.Write("Input the marks obtained in Maths: ");
            int maths = Convert.ToInt32(Console.ReadLine());

            Console.Write("Input the marks obtained in Physics: ");
            int physics = Convert.ToInt32(Console.ReadLine());

            Console.Write("Input the marks obtained in Chemistry: ");
            int chemistry = Convert.ToInt32(Console.ReadLine());

            // Check eligibility
            if (maths >= 65 && physics >= 55 && chemistry >= 50 &&
                (maths + physics + chemistry >= 180 || maths + physics >= 140))
            {
                Console.WriteLine("The candidate is eligible for admission.");
            }
            else
            {
                Console.WriteLine("The candidate is not eligible for admission.");
            }
```

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

```
D:\.NET\Ayushi_Tutorials>T2Q14
Input the marks obtained in Maths: 72
Input the marks obtained in Physics: 65
Input the marks obtained in Chemistry: 51
The candidate is eligible for admission.
```

15.Write a Program which accepts name from the user and prints the same

INPUT : R K University
OUTPUT: R K University

```csharp
using System;

namespace Ayushi_Tutorials
{
    internal class T2Q15
    {
        static void Main(string[] args)
        {
            // Ask user for input
            Console.Write("Enter your name: ");
            string name = Console.ReadLine();  // Read entire line including spaces

            // Print the name
            Console.WriteLine("Output: " + name);

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

```
D:\.NET\Ayushi_Tutorials>T2Q15
Enter your name: Ayushi
Output: Ayushi
```