**Task-2**

1 Write a C# Sharp program to find out whether a given year is a leap year or not. Test Data: 2016

1. Write a C# Sharp program to check whether a triangle is Equilateral, Isosceles or Scalene. Test Data :50 50 60
2. Write a C# Sharp program to calculate and print the electricity bill of a customer. From the keyboard, the customer's name, ID, and unit consumed should be taken and displayed along with the total amount due.

The charge are as follow :

|  |  |
| --- | --- |
| **Unit** | **Charge/unit** |
| upto 199 | 1.20 |
| 200 and above but less than 400 | 1.50 |
| 400 and above but less than 600 | 1.80 |
| 600 and above | 2.00 |

If bill exceeds Rs. 400 then a surcharge of 15% will be charged and the minimum bill should be of Rs. 100/-

Test Data :

1001

James 800

1. Write a C# Sharp program that takes three letters and displays them in reverse order.

Test Data: Enter letter: b

Enter letter: a Enter letter: t

Expected Output: t a b

1. Write a program in C# Sharp to compare two strings. Test Data: Input the 1st string: This is first string

Input the 2nd string: This is first string.

1. Write a program in C# Sharp to check the username and password.

Test Data :Input a username: user Input a password: pass

Input a username: abcd Input a password: 1234

Expected Output :Password entered successfully!

1. Write a C# Sharp program that takes user id and password as input (string type). After 3 unsuccessful attempts, the user will be rejected.

# Note : when you are creating matrix it should be of 3 x 3

1. Write a program to create the menu driven application using c#. Menus are as given below
   1. Square of created matrix.
   2. Create another matrix and perform addition with the above matrix.
   3. Write a program to perform multiplication of the new matrix with the above matrix.
   4. Print transpose of matrix.
2. Write a program to create the menu driven application using c#.Menus are as given below :
   1. Print diagonals of matrix
   2. Print upper triangular matrix
   3. Print lower triangular matrix
   4. Print determinant of matrix
3. Write a C# Sharp program in to count duplicate elements in an array.
4. Write a program in C# Sharp to separate odd and even integers into separate arrays. Test Data: Input the number of elements to be stored in the array :5

Input 5 elements in the array: element - 0: 25

element - 1: 47

element - 2: 42

element - 3: 56

element - 4: 32

1. Write a program to create the menu driven application using different types of looping statements. Menus are as given below.
   1. Get the even natural number from the user print the sum of even natural numbers. 2.Get the number from user and print the table of given number, Example: 5 X 1 = 5 3.Write a program to find the given number is perfect or not

4. Write a program to find if the given number is strong or not.

1. Write a C# Sharp program to create a structure that stores two data for an employee.

*Test Data*: Name of the employee: H. Rana Input day of the birth: 05

Input month of the birth: 02 Input year for the birth: 58

1. Write a program in C# Sharp to insert two books' information.

*Test Data* :Insert the information of two books : Information of book 1

Input name of the book: BASIC Input the author: S.TROELSTRA

1. Write a program in C# Sharp to implement a structure named Square. Test Data :Input the dimensions of the Square(equal length and breadth ) :

length: 20

breadth: 40

1. Write a program to create an enumeration of “Months” and print the same using for-each loop.
2. Write a program to create an enumeration of “Status” and print using switch case. Status are like: open, inProgress, resolved, closed, reopened
3. Write a C# program to implement a method that takes an integer as input and throws an exception if the number is negative.
4. Write a C# program that implements a method that divides two numbers. Handle the DivideByZeroException that occurs if the denominator is 0.
5. Write a program to create a menu driven application. Menus are as given below.

1.

\*

\*\*

\*\*\*

\*\*\*\* 2.\*

\* \*

\* \* \*

\* \* \* \*

3. \*

\*\*

\* \* \*

\* \* \* \*

\* \* \* \* \*

4.

\*

\*\*\*

\*\*\*\*\*

\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*

\*\*\*\*\*

\*\*\*

\*