.Net Technology (01CE0602)

Department of Computer Engineering

6th Semester

Lab Manual

(Year: 2022-23)

Index

Lab	Programs	Date	Signature
1	a. Write a C# Program to print "Hello World". b. Write a C# Program to add 2 Numbers. c. Write a C# Program to find maximum of 2 Numbers. d. Write a C# Program to generate electricity bill using Else-If ladder. e. Write a C# Program to find the sum of first N numbers.		
2	a. Write a C# Program to check a number is Palindrome or not. b. Write a C# Program to generate Fibonacci series up to N Numbers. c. Write a C# program to create a calculator using Switch Case. d. Write a C# Program to print a given array in reverse. e. Create a simple C# code for the following: 55555 4444 333 22 1		
3	a. Write a C# program find area of Circle, Rectangle and Square using Polymorphism. b. Consider a class Information that has data members as Name, Surname and Contact number. Let Employee and Student class inherits Information class with its own other information such as Students Semester or Employee Salary. Implement a system using Method Overriding to take input from the user for all the information and display proper output. c. Consider a class Apartment that has data members as Apartment number and balcony type. Implement a system that has 3 classes as 1bhk, 2bhk and 3bhk such that it does not allow to create any other classes above 3bhk. Also implement inheritance in such a way that 1bhk will have Rectangular Balcony and all other flats have Rounded Balcony (Use Sealed Class).		
4	a. Apply Interface to find the area of Square, Rectangle and Circle. Display proper output. b. Create two interfaces Icredit and Idebit with methods deposit and withdraw respectively. Create a class Account that inherits interface such that it provides the functionality of Crediting some amount and withdrawing some amount. Use Proper Variables and display output accordingly. c. Demonstrate a calculator using delegate.		

<table>
<tr><td>5</td><td>a. Write a C# Console based application to create following table using ADO. Net.
b. Write a C# Console based application to display all the records of a table.
<table>
<tr><th>Emp_id</th><th>Name</th><th>Designation</th><th>Department</th><th>Salary</th></tr>
<tr><td>1</td><td>Raj</td><td>Manager</td><td>Sales</td><td>35000</td></tr>
<tr><td>2</td><td>Priya</td><td>Manager</td><td>HR</td><td>30000</td></tr>
<tr><td>3</td><td>Manoj</td><td>Driver</td><td>Transport</td><td>15000</td></tr>
<tr><td>4</td><td>Aakash</td><td>Executive</td><td>Finance</td><td>85000</td></tr>
</table></td><td></td><td></td></tr>
<tr><td>6</td><td>a. Write a C# Console based application to implement a functionality to insert a new record in the table
b. Write a C# Console based application to implement a functionality to display specific record from the table
Use Above Table as per Lab 5.</td><td></td><td></td></tr>
<tr><td>7</td><td>a. Create a Simple Calculator using Windows Forms.
b. Create a Windows Forms that will change the background color, forecolor and styling of the given text.</td><td></td><td></td></tr>
<tr><td>8</td><td>a. Create a Windows Form that will move the data from one tool to other tool (Usage of ComboBox and ListBox)
b. Create a GUI for the following: Consider textbox(txt1) for Full Name, textbox(txt2) for enrolment, textbox(txt3) for email, textbox(txt4) for mobile, combobox(cmb1) for Semester, radiobutton(rd1,rd2) for Gender and datetimepicker(dtp1) for birthdate and button(btn1). Write backend code for taking input of each and display all values in pop-up box with proper message on button (btn1) click.</td><td></td><td></td></tr>
<tr><td>9</td><td>a. Create a GUI for the following: Consider textbox(txt1) for Full Name, textbox(txt2) for enrolment, textbox(txt3) for email, textbox(txt4) for mobile, textArea(txtarea1) for Address, textbox(txt5) for City, combobox(cmb1) for Semester, radiobutton(rd1,rd2) for Gender and datetimepicker(dtp1) for birthdate, checkbox(ck1) for Agree to Register and button(btn1). Write backend code for taking input of each control and if Agree checkbox is checked, then store all these data in the database. Show Pop-Up message: “Registration Successful”.</td><td></td><td></td></tr>
<tr><td>10</td><td>a. Write a Windows based application to create following table using ADO. Net.
b. Write a C# Windows based application to display all the records of a table.
Use Above Table as per Lab 5.</td><td></td><td></td></tr>
<tr><td>11</td><td>a. Write a C# Windows based application to implement a functionality to insert a new record in the table
b. Write a C# Windows based application to implement a functionality to display specific record from the table.
Use Above Table as per Lab 5.</td><td></td><td></td></tr>
</table>

12	a. Write a C# Windows based application to implement a functionality to insert a new record in the table b. Write a C# Windows based application to implement a functionality to display specific record from the table. Use Above Table as per Lab 5.		
13	a. Write a C# code to generate 3 different lines of different colors. b. Write a C# code to generate 4 different lines of Multicolor Rectangle. c. Write a C# code to generate 2 ellipses on a windows form.		
14	a. Write XAML code for the following: Consider textbox(txt1) for First Number, textbox(txt2) for Second Number, textbox(txt3) for Answer, and 4 buttons (btn1,btn2,btn3,btn4) for Addition, Subtraction, Multiplication and Division respectively. Write backend code for taking input of 2 numbers and display relevant output as per button click. b. Write XAML code for the following: Consider textbox(txt1) for Full Name, textbox(txt2) for enrolment, textbox(txt3) for email, textbox(txt4) for mobile, combobox(cmb1) for Semester, radiobutton(rd1,rd2) for Gender and datetimepicker(dtp1) for birthdate and button(btn1). Write backend code for taking input of each and display all values in pop-up box with proper message on button (btn1) click.		

LAB - 1

Program 1: Write a C# Program to print “Hello World”.

Code:

```
1   using System;
2
3   namespace HelloWorld
4   {
        0 references
5       internal class Program
6       {
            0 references
7           private static void Main(string[] args)
8           {
9               Console.WriteLine("HELLO!");
10              Console.WriteLine("92000103073-Raj Chhadia");
11              Console.Read();
12          }
13      }
14  }
```

Output:

```
D:\Marwadi\SEM-6\dotNET\LAB\Practical 1\Addition\HelloV
HELLO!
92000103073-Raj Chhadia
```

Program 2: Write a C# Program to add 2 Numbers.

Code:

```
namespace Addition
{
    using System;

    0 references
    internal class Program
    {
        0 references
        private static void Main(string[] args)
        {
            Console.WriteLine("Addtion of two numbers");
            Console.WriteLine("Enter 1st INTEGER");
            int a = int.Parse(Console.ReadLine());
            Console.WriteLine("Enter 2nd INTEGER");
            int b = int.Parse(Console.ReadLine());

            int c = a + b;

            Console.WriteLine("Answer: " + c);
            Console.Read();
        }
    }
}
```

Output:

```
D:\Marwadi\SEM-6\dotNET\LAB\Practical 1\Addition\Addition\bin\De
Addtion of two numbers
Enter 1st INTEGER
15
Enter 2nd INTEGER
20
Answer: 35
```

Program 3: Write a C# Program to find maximum of 2 Numbers.

Code:

```
1   using System;
2
3   namespace MaxOfTwo
4   {
        0 references
5       internal class Program
6       {
            0 references
7           private static void Main(string[] args)
8           {
9               Console.WriteLine("Max of two numbers");
10              Console.WriteLine("Enter 1st INTEGER");
11              int a = int.Parse(Console.ReadLine());
12              Console.WriteLine("Enter 2nd INTEGER");
13              int b = int.Parse(Console.ReadLine());
14              if (a > b)
15                  Console.WriteLine("Max is : " + a);
16              else if (b > a)
17                  Console.WriteLine("Max is: " + b);
18              else
19                  Console.WriteLine("Both are equal");
20              Console.Read();
21          }
22      }
23  }
```

Output:

```
D:\Marwadi\SEM-6\dotNET\LAB\Practical 1\Addition\MaxOfTwo\bin\Debug\Ma
Max of two numbers
Enter 1st INTEGER
5
Enter 2nd INTEGER
69
Max is: 69
```

Program 4: Write a C# Program to generate electricity bill using Else-If ladder

Code:

```
using System;

namespace ElectricityBill
{
    0 references
    internal class Program
    {
        0 references
        private static void Main(string[] args)
        {
            Console.WriteLine("Electricity Bill Calaculator");
            Console.WriteLine("Enter number of units consumed: ");
            float a = float.Parse(Console.ReadLine());
            if (a > 0 && a <= 100)
                Console.WriteLine("Bill amount is : " + (a * 5));
            else if (a >= 100 && a < 200)
                Console.WriteLine("Bill amount is : " + (a * 10));
            else if (a >= 200 && a < 300)
                Console.WriteLine("Bill amount is : " + (a * 20));
            else if (a >= 300 && a < 400)
                Console.WriteLine("Bill amount is : " + (a * 30));
            else if (a >= 400 && a < 500)
                Console.WriteLine("Bill amount is : " + (a * 40));
            else
                Console.WriteLine("Bill amount is : " + (a * 50));
            Console.Read();
        }
    }
}
```

Output:

```
D:\Marwadi\SEM-6\dotNET\LAB\Practical 1\Addition\ElectricityBill\bin\Debug\ElectricityBil
Electricity Bill Calaculator
Enter number of units consumed:
55
Bill amount is : 275
```

Program 5: Write a C# Program to find the sum of first N numbers.

Code:

```
using System;

namespace SumOfN
{
    0 references
    internal class Program
    {
        0 references
        private static void Main(string[] args)
        {
            Console.WriteLine("Sum of n numbers");
            Console.WriteLine("Enter value of n");
            int a = int.Parse(Console.ReadLine());
            int sum = 0;

            for (int i = 1; i <= a; i++)
            {
                sum += i;
            }
            Console.WriteLine("Sum is: " + sum);
            Console.Read();
        }
    }
}
```

Output:

```
D:\Marwadi\SEM-6\dotNET\LAB\Practical 1\Additi
Sum of n numbers
Enter value of n
10
Sum is: 55
```

LAB - 2

Program 1: Write a C# Program to check a number is Palindrome or not.

Code:

```
using System;

namespace Palindrome
{
    0 references
    internal class Program
    {
        0 references
        private static void Main(string[] args)
        {
            Console.WriteLine("Palindrome checker");
            Console.WriteLine("Enter the string: ");
            string originalString = Console.ReadLine();
            char[] stringArray = originalString.ToCharArray();
            Array.Reverse(stringArray);
            string reverseString = new string(stringArray);

            if (reverseString.Equals(originalString))
                Console.WriteLine("String is Palindrome");
            else
                Console.WriteLine("String is not Palindrome");
            Console.Read();
        }
    }
}
```

Output:

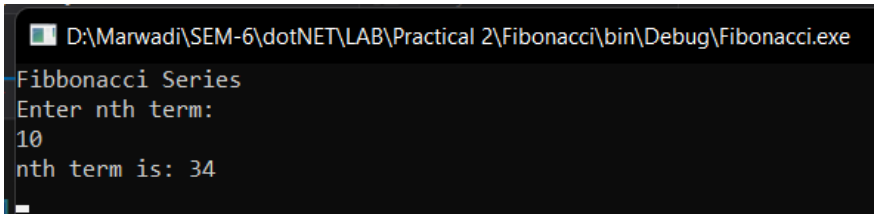
```
D:\Marwadi\SEM-6\dotNET\LAB\Practical 2\Palindrome\Palindrome\b
Palindrome checker
Enter the string:
exe
String is Palindrome
```

Program 2: Write a C# Program to generate Fibonacci series up to N Numbers.

Code:

```
using System;

namespace Fibonacci
{
    0 references
    internal class Program
    {
        3 references
        public static int Fibonacci(int n)
        {
            if (n == 1)
                return 0;
            else if (n == 2)
                return 1;
            else
                return Fibonacci(n - 1) + Fibonacci(n - 2);
        }

        0 references
        private static void Main(string[] args)
        {
            Console.WriteLine("Fibbonacci Series");
            Console.WriteLine("Enter nth term: ");
            int number = int.Parse(Console.ReadLine());
            int term = Fibonacci(number);
            Console.WriteLine("nth term is: " + term);
            Console.Read();
        }
    }
}
```

Output:



Program 3: Write a C# program to create a calculator using Switch Case.

Code:

```
using System;

namespace Calculator
{
    0 references
    internal class Program
    {
        0 references
        private static void Main(string[] args)
        {
            Console.WriteLine("Calculator");
            Console.WriteLine("Enter operation (+.-,*,/): ");
            string operation = Console.ReadLine();
            Console.WriteLine("Enter 1st number: ");
            float a = float.Parse(Console.ReadLine());
            Console.WriteLine("Enter 2nd number: ");
            float b = float.Parse(Console.ReadLine());
            switch (operation)
            {
                case "+":
                    Console.WriteLine("Addition of two number is: " + (a + b));
                    break;

                case "-":
                    Console.WriteLine("Subtraction of two number is: " + (a - b));
                    break;

                case "*":
                    Console.WriteLine("Multiplication of two number is: " + (a * b));
                    break;

                case "/":
                    Console.WriteLine("Division of two number is: " + (a / b));
                    break;

                default:
                    Console.WriteLine("Invalid Input!");
                    break;
            }
            Console.Read();
        }
    }
}
```

Output:

```
D:\Marwadi\SEM-6\dotNET\LAB\Practical 2\Calaculator\bin
Calculator
Enter operation (+.-,*,/):
*
Enter 1st number:
5
Enter 2nd number:
6
Multiplication of two number is: 30
```

Program 4: Write a C# Program to print a given array in reverse.

Code:

```
using System;

namespace Reverse
{
    0 references
    internal class Program
    {
        0 references
        private static void Main(string[] args)
        {
            Console.WriteLine("Reversal of array");
            Console.WriteLine("Enter the number of element : ");
            int size = Convert.ToInt32(Console.ReadLine());

            int[] original_array = new int[size];

            for (int i = 0; i < size; i++)
            {
                Console.Write("Array[{0}] :", i);
                original_array[i] = Convert.ToInt32(Console.ReadLine());
            }
            Array.Reverse(original_array);
            for (int i = 0; i < size; i++)
            {
                Console.Write("Reverse array is :");
                Console.WriteLine(original_array[i]);
            }
            Console.Read();
        }
    }
}
```

Output:

```
D:\Marwadi\SEM-6\dotNET\LAB\Practical 2\Reverse\bin\Debug\
Reversal of array
Enter the number of element :
3
Array[0] :0
Array[1] :7
Array[2] :3
Reverse array is :3
Reverse array is :7
Reverse array is :0
```

Program 5: Create a simple C# code for the following:

55555

4444

333

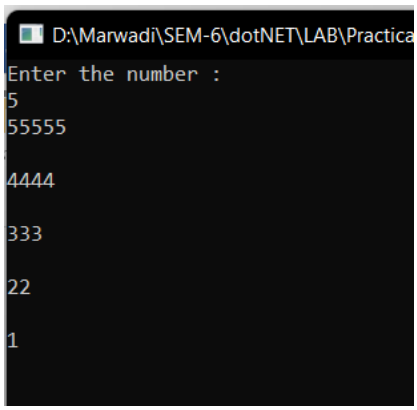
22

1

Code:

```
using System;

namespace StarPattern
{

    internal class StarPattern
    {
        0 references
        private static void Main()
        {
            Console.WriteLine("Enter the number : ");
            int number = Convert.ToInt32(Console.ReadLine());

            for (int i = 0; i < number; i++)
            {
                for (int j = number - i; j > 0; j--)
                {
                    Console.Write(number - i);
                }
                Console.WriteLine("\n");
            }
            Console.Read();
        }
    }
}
```

Output:



LAB - 3

Program 1: Write a C# program find area of Circle, Rectangle and Square using Polymorphism.

Code:

```
using System;

namespace Area
{
    3 references
    internal class Circle
    {
        1 reference
        public void area(float r)
        {
            float area = (float)3.14 * r * r;
            Console.WriteLine("Area of Circle is: " + area);
        }
    }

    3 references
    internal class Rectangle : Circle
    {
        1 reference
        public void area(float l, float b)
        {
            float area = (float)l * b;
            Console.WriteLine("Area of Rectangle is: " + area);
        }
    }

```

```
23      internal class Square : Rectangle
24      {
            1 reference
25          public void area(float s)
26          {
27              float area = (float)s * s;
28              Console.WriteLine("Area of Square is: " + area);
29          }
30      }
31
            0 references
32      internal class Program : Square
33      {
            0 references
34          private static void Main(string[] args)
35          {
36              Circle c = new Circle();
37              Rectangle r = new Rectangle();
38              Square s = new Square();
39              Console.WriteLine("Raj Chhadia");
40              Console.WriteLine("Enter the radius of circle: ");
41              c.area(float.Parse(Console.ReadLine()));
42              Console.WriteLine("Enter the Length and breadth of Rectangle: ");
43              r.area(float.Parse(Console.ReadLine()), float.Parse(Console.ReadLine()));
44              Console.WriteLine("Enter the side of square: ");
45              s.area(float.Parse(Console.ReadLine()));
46              Console.Read();
47          }
48      }
49  }
```

Output:

```
D:\Marwadi\SEM-6\dotNET\LAB\Practical 3\Area\Area\bin\Debug\Area.
Raj Chhadia
Enter the radius of circle:
10
Area of Circle is: 314
Enter the Length and breadth of Rectangle:
5
6
Area of Rectangle is: 30
Enter the side of square:
8
Area of Square is: 64
```

Program 2: Consider a class Information that has data members as Name, Surname and Contact number. Let Employee and Student class inherits Information class with its own other information such as Students Semester or Employee Salary. Implement a system using Method Overriding to take input from the user for all the information and display proper output.

Code:

```
using System;

namespace Information
{
    0 references
    internal class Program
    {
        0 references
        private static void Main(string[] args)
        {
            Console.WriteLine("Raj Chhadia");
            Student s1 = new Student();
            Employee e1 = new Employee();
            Console.WriteLine("Enter your choice: ");
            Console.WriteLine("1. Student");
            Console.WriteLine("2. Employee");
            int choice = int.Parse(Console.ReadLine());
            switch (choice)
            {
                case (1):
                    s1.getData();
                    s1.putData();
                    break;

                case (2):
                    e1.getData();
                    e1.putData();
                    break;
            }
            Console.Read();
```

```
29          }
30      }
31
        2 references
32      internal class Information
33      {
34          public string name;
35          public string surname;
36          public int number;
37
            2 references
38          public void getData()
39          {
40              Console.WriteLine("Enter the first name: ");
41              name = Console.ReadLine();
42              Console.WriteLine("Enter the surname: ");
43              surname = Console.ReadLine();
44              Console.WriteLine("Enter the contact number: ");
45              number = int.Parse(Console.ReadLine());
46          }
47      }
48
        2 references
49      internal class Employee : Information
50      {
51          private int salary;
52
            1 reference
53          public void getData()
54          {
55              base.getData();
```

```
56              Console.WriteLine("Enter the employee salary: ");
57              salary = int.Parse(Console.ReadLine());
58          }
59
60          public void putData()
61          {
62              Console.WriteLine("Name: " + name);
63              Console.WriteLine("SurName: " + surname);
64              Console.WriteLine("Contact Number: " + number);
65              Console.WriteLine("Salary: " + salary);
66          }
67      }
68
69      internal class Student : Information
70      {
71          private int semester;
72
73          public void getData()
74          {
75              base.getData();
76              Console.WriteLine("Enter the semester number: ");
77              semester = int.Parse(Console.ReadLine());
78          }
79
```

```
1 reference
80      public void putData()
81      {
82          Console.WriteLine("Name: " + name);
83          Console.WriteLine("SurName: " + surname);
84          Console.WriteLine("Contact Number: " + number);
85          Console.WriteLine("Semester: " + semester);
86      }
87  }
88  }
```

Output:

```
Raj Chhadia
Enter your choice:
1. Student
2. Employee
1
Enter the first name:
Raj
Enter the surname:
C
Enter the contact number:
12
Enter the semester number:
6
Name: Raj
SurName: C
Contact Number: 12
Semester: 6
```

Program 3: Consider a class Apartment that has data members as Apartment number and balcony type. Implement a system that has 3 classes as 1bhk,2bhk and 3bhk such that it does not allow to create any other classes above 3bhk. Also implement inheritance in such a way that 1bhk will have Rectangular Balcony and all other flats have Rounded Balcony (Use Sealed Class).

Code:

```
using System;

namespace Apartment
{
    0 references
    internal class Program
    {
        0 references
        private static void Main(string[] args)
        {
            OneBHK f1 = new OneBHK();
            f1.welcome();
            f1.display();
            f1.balcony();

            TwoBHK f2 = new TwoBHK();
            f2.welcome();
            f2.display();
            f2.balcony();

            ThreeBHK f3 = new ThreeBHK();
            f3.welcome();
            f3.display();
            f3.balcony();

            Console.Read();
        }
    }

```

```
28      public class Flat
29      {
            3 references
30          public void welcome()
31          {
32              Console.WriteLine("Welcome to Beautiful Apartment... Raj Chhadia");
33          }
34
            6 references
35          public virtual void display()
36          {
37              Console.WriteLine("Not ready yet...");
38          }
39
            3 references
40          public virtual void features()A
41          {
42              Console.WriteLine("Not ready yet...");
43              balcony();
44          }
45
            9 references
46          public virtual void balcony()
47          {
48              Console.WriteLine("Not ready yet...");
49          }
50      }
51
```

```
52      public class OneBHK : Flat
53      {
            6 references
54          public override void display()
55          {
56              Console.WriteLine("This is 1 BHK flat");
57          }
58
            3 references
59          public override void features()
60          {
61              Console.WriteLine("\tFeatures...");
62              balcony();
63          }
64
            9 references
65          public override void balcony()
66          {
67              Console.WriteLine("\tBalcony: Rectangular\n");
68          }
69      }
70
        3 references
71      public class TwoBHK : OneBHK
72      {
            6 references
73          public override void display()
74          {
75              Console.WriteLine("This is 2 BHK flat");
76          }
```

```
78          public override void features()
79          {
80              Console.WriteLine("\tFeatures...");
81              balcony();
82          }
83
                9 references
84          public override sealed void balcony()
85          {
86              Console.WriteLine("\tBalcony: Circular\n");
87          }
88      }
89
            2 references
90      public sealed class ThreeBHK : TwoBHK
91      {
                6 references
92          public override void display()
93          {
94              Console.WriteLine("This is 3 BHK flat");
95          }
96
                3 references
97          public override void features()
98          {
99              Console.WriteLine("\tFeatures...");
100             balcony();
101         }
102     }
```

Output:

```
D:\Marwadi\SEM-6\dotNET\LAB\Practical 3\Apartment\bin\Debug\Apartment.exe
Welcome to Beautiful Apartment... Raj Chhadia
This is 1 BHK flat
        Balcony: Rectangular

Welcome to Beautiful Apartment... Raj Chhadia
This is 2 BHK flat
        Balcony: Circular

Welcome to Beautiful Apartment... Raj Chhadia
This is 3 BHK flat
        Balcony: Circular
```

LAB - 4

Program 1: Apply Interface to find the area of Square, Rectangle and Circle. Display proper output.

Code:

```
namespace Interface_area
{
    using System;

    3 references
    internal interface Area
    {
        6 references
        void area();
    }

    3 references
    internal class Square : Area
    {
        private float side;

        1 reference
        public Square(float side)
        {
            this.side = side;
            this.area();
        }

        2 references
        public void area()
        {
            Console.WriteLine("Area of sqaure is: " + (side * side));
        }
    }

    internal class Rectangle : Area
    {
        private float length;

        private float breadth;

        1 reference
        public Rectangle(float length, float breadth)
        {
            this.length = length;
            this.breadth = breadth;
            this.area();
        }

        2 references
        public void area()
        {
            Console.WriteLine("Area of rectangle is: " + (length * breadth));
        }
    }

}
```

```
3 references
internal class Circle : Area
{
    private float radius;

    1 reference
    public Circle(float radius)
    {
        this.radius = radius;
        this.area();
    }

    2 references
    public void area()
    {
        Console.WriteLine("Area of circle is: " + (3.14 * radius * radius));
    }
}
```

```
0 references
internal class Program
{
    0 references
    private static void Main(string[] args)
    {
        Console.WriteLine("Area using interface");
        Console.WriteLine("Raj Chhadia");
        float side, length, breadth, radius;

        Console.WriteLine("Enter side of a square");
        side = float.Parse(Console.ReadLine());
        Square sq = new Square(side);

        Console.WriteLine("Enter length of a rectangle");
        length = float.Parse(Console.ReadLine());
        Console.WriteLine("Enter breadth of a rectangle");
        breadth = float.Parse(Console.ReadLine());
        Rectangle rec = new Rectangle(length, breadth);

        Console.WriteLine("Enter radius of a circle");
        radius = float.Parse(Console.ReadLine());
        Circle circ = new Circle(radius);

        Console.Read();
    }
}
```

Output:

```
D:\Marwadi\SEM-6\dotNET\LAB\Practical 4\Interface_area\Interface_area\bin\Debug\Interface_area.exe
Area using interface
Raj Chhadia
Enter side of a square
5
Area of sqaure is: 25
Enter length of a rectangle
6
Enter breadth of a rectangle
7
Area of rectangle is: 42
Enter radius of a circle
10
Area of circle is: 314
```

Program 2: Create two interfaces Icredit and Idebit with methods deposit and withdraw respectively. Create a class Account that inherits interface such that it provides the functionality of Crediting some amount and withdrawing some amount. Use Proper Variables and display output accordingly.

Code:

```
using System;

namespace bank
{
    1 reference
    internal interface Icredit
    {
        2 references
        void deposit();
    }

    1 reference
    internal interface Idebit
    {
        2 references
        void withdraw();
    }

    internal class Account : Icredit, Idebit
    {
        private int balance = 0;

        2 references
        public void deposit()
        {
            Console.WriteLine("\nEnter amount to deposit : ");
            balance += int.Parse(Console.ReadLine());
            Console.WriteLine("Deposite successful ");
            Console.WriteLine("New balance is : " + balance);
        }

        2 references
        public void withdraw()
        {
            Console.WriteLine("\nEnter amount to withdraw : ");
            int amount = int.Parse(Console.ReadLine());

            if (amount > balance)
            {
                Console.WriteLine("Can not be withdraw ");
            }
            else
            {
                Console.WriteLine("withdraw successful ");
                balance -= amount;
                Console.WriteLine("New balance is : " + balance);
```

```
41              }
42          }
43      }
44
        0 references
45      internal class Program
46      {
            0 references
47          private static void Main(string[] args)
48          {
49              Console.WriteLine("\nEnter your choice : ");
50              Console.WriteLine("1. deposit : ");
51              Console.WriteLine("2. withdraw : ");
52              Console.WriteLine("3. exit : ");
53              int choice = int.Parse(Console.ReadLine());
54
55              Account a = new Account();
56
57              while (choice != 3)
58              {
59                  if (choice == 1)
60                  {
61                      a.deposit();
62                  }
63                  else if (choice == 2)
64                  {
65                      a.withdraw();
66                  }
67                  Console.WriteLine("\nEnter your choice : ");
68                  Console.WriteLine("1. deposit : ");
69                  Console.WriteLine("2. withdraw : ");
70                  Console.WriteLine("3. exit : ");
71                  choice = int.Parse(Console.ReadLine());
72                  Console.Read();
73              }
74          }
75      }
76  }
```

Output:

```
D:\Marwadi\SEM-6\dotNET\LAB\Practical 4\Credit_Debit\bin\Debug\Credit_Debit.exe
1. deposit :
2. withdraw :
3. exit :

Enter your choice :
1

Enter amount to deposit :
500
Deposite successful
New balance is : 500
1. deposit :
2. withdraw :
3. exit :

Enter your choice :
2

Enter amount to withdraw :
800
Can not be withdraw
1. deposit :
2. withdraw :
3. exit :

Enter your choice :
3
```

Program 3 Demonstrate a calculator using delegate

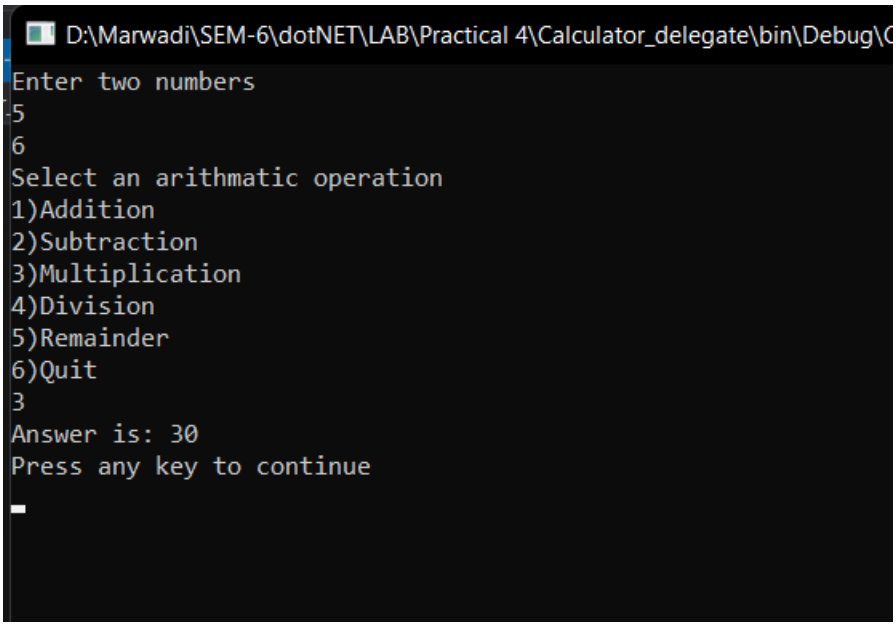
Code:

```
using System;

namespace ArithmaticOperation
{
    0 references
    class Program
    {
        delegate double ArithmaticDelegate(double a, double b);

        1 reference
        static void Menu()
        {
            Console.WriteLine("Select an arithmatic operation");
            Console.WriteLine("1)Addition");
            Console.WriteLine("2)Subtraction");
            Console.WriteLine("3)Multiplication");
            Console.WriteLine("4)Division");
            Console.WriteLine("5)Remainder");
            Console.WriteLine("6)Quit");
        }

        1 reference
        static double Add(double a, double b)
        {
            return a + b;
        }

        1 reference
        static double Subtract(double a, double b)
        {
            return a - b;
        }

        static double Multiply(double a, double b)
        {
            return a * b;
        }

        1 reference
        static double Divide(double a, double b)
        {
            return a / b;
        }

        1 reference
        static double Modulus(double a, double b)
        {
            return a % b;
        }
```

```
45          static void Main(string[] args)
46          {
47              int operation;
48              ArithmaticDelegate arithmatic = null;
49              double n1, n2;
50
51              do
52              {
53                  Console.WriteLine("Enter two numbers");
54                  n1 = double.Parse(Console.ReadLine());
55                  n2 = double.Parse(Console.ReadLine());
56
57                  Menu();
58                  operation = int.Parse(Console.ReadLine());
59
60                  switch (operation)
61                  {
62                      //Addition
63                      case 1:
64                          arithmatic = new ArithmaticDelegate(Add);
65                          break;
66                      //Subtraction
67                      case 2:
68                          arithmatic = new ArithmaticDelegate(Subtract);
69                          break;
70                      //Multiplication
71                      case 3:
72                          arithmatic = new ArithmaticDelegate(Multiply);
73                          break;
74                      //Division
```

```
75                      case 4:
76                          arithmatic = new ArithmaticDelegate(Divide);
77                          break;
78                      //Remainder
79                      case 5:
80                          arithmatic = new ArithmaticDelegate(Modulus);
81                          break;
82                      default:
83                          Console.WriteLine("Exiting program");
84                          break;
85                  }
86                  if (arithmatic != null)
87                  {
88                      Console.WriteLine("Answer is: "+arithmatic(n1, n2));
89                  }
90                  Console.WriteLine("Press any key to continue");
91                  Console.ReadKey(true);
92                  Console.Clear();
93
94              } while (operation != 6);
95
96          }
97      }
98  }
```

Output:


D:\Marwadi\SEM-6\dotNET\LAB\Practical 4\Calculator_delegate\bin\Debug\C
Enter two numbers
5
6
Select an arithmatic operation
1)Addition
2)Subtraction
3)Multiplication
4)Division
5)Remainder
6)Quit
3
Answer is: 30
Press any key to continue