

Chapter # 3

Sr.no	Topic	Date	Sign
1.	Reading strings from the keyboard	9-1-2007	
2.	Changing String order	9-1-2007	
3.	More than one class	9-1-2007	
4.	Assigning values to variables	9-1-2007	
5.	Diamond pattern on the console screen	9-1-2007	

Chapter # 4

Sr.no	Topic	Date	Sign
1.	Illustrating the Concept of Declaration of variables	16-1-2007	
2.	Declaration & Additions of variables	16-1-2007	
3.	Program with a function	16-1-2007	
4.	Demonstrating Boxing & Unboxing	16-1-2007	
5.	Demonstrating addition of byte type variables	16-1-2007	
6.	Implementing some custom console output	16-1-2007	
7.	Printing a home like figure in the console	16-1-2007	
8.	Executing some console statements	16-1-2007	

Chapter # 3

(Overview of C#)

3.1 - Reading strings from the keyboard

```
using System;

class Prog3_1
{
    public static void Main()
    {
        Console.Write ("Enter Your First Name : "); // Displaying to write first name
        string name1 = Console.ReadLine (); // Saving first name in name1

        Console.Write ("Enter Your Last Name : "); // Displaying to write last name
        string name2 = Console.ReadLine (); // Saving first name in name2

        Console.WriteLine ("Hello Mr." + name1 + " " + name2); // Displaying both first & last names

        Console.ReadLine (); // Since to stop the console for displaying last line, we use this to accept a
        keystroke frm user. (Similar to getch() in C)
    }
}
```

OUTPUT

Enter Your First Name: Daljit

Enter Your Last Name: Singh

Hello Mr. Daljit Singh

3.2 - Changing String order

```
using System;
```

```
class Prog3_2
```

```
{
```

```
    public static void Main(String [] args)
```

```
    {
```

```
        Console.Write(args[2] + args[0] + args[1]);
```

```
    }
```

```
}
```

3.3 – More than one class

```
using System;

class ClassOne
{
    public void One() // A function named One
    {
        Console.Write("C Sharp ");
    }
}

class Mainly
{
    public static void Main() // A function named Main (Main Function)
    {
        ClassOne demoObj = new ClassOne (); //Creating ojecct of ClassOne
        demoObj.One (); // Will display ---> C Sharp
        Console.Write ("Programming"); // Will display ---> Programming
        // Both "C Sharp" & "Programming" will be displayed in a single line ---->
        Console.Write("C Sharp ");
        Console.ReadLine ();
    }
}
```

OUTPUT

C Sharp Programming

3.4 – Assigning values to variables

```
using System;

class SampleMath
{
    public static void Main()
    {
        double x = 2.0; // declaring a variable named x of type double & assigning it value 2.0
        double y = 3.0; // declaring a variable named y of type double & assigning it value 3.0
        double z; // declaring a variable named z of type double
        z = x + y;
        Console.WriteLine("x = " + x + ", y = " + y + " & z = " + z);
        Console.ReadLine();
    }
}
```

OUTPUT

X = 2.0, Y = 3.0, Z = 5.0

3.5 – Diamond pattern on the console screen

```
using A = System.Console;

class Pattern
{
    public static void Main()
    {
        A.WriteLine (" X ");
        A.WriteLine (" XXX ");
        A.WriteLine ("XXXXX");
        A.WriteLine (" XXX ");
        A.WriteLine (" X ");
        A.ReadLine ();
    }
}
```

OUTPUT

```
X
XX
XXX
XX
X
```

Chapter # 4

(Literals, Variables & Data Types)

4.1 – Illustrating the Concept of Declaration of variables

```
class Variable_Concepts
{
    public static void Main ()
    {
        char ch = 'A'; // Declaring a Character variable with value = 'A'
        byte a = 50; // Declaring a byte variable with value = 50
        int b = 123456789; // Declaring an Integer variable with value = 123456789
        long c = 1234567654321; // Declaring a Long type variable with value = 1234567654321
        bool d = true; // Declaring a Boolean type variable with TRUE value
        float e = 0.000000345F; // Declaring a float type variable with value = 0.000000345. The value ends with a
        'F' resembling a float data type
        float f = 1.23e5F; // Declaring a float type exponential variable with value = 1.23E5 = 123000. The value
        contains the character 'e' resembling an exponential value. Also, the value ends with a 'F' resembling a float
        data type.
    }
}
```

4.2 – Declaration & Additions of variables

using System;

class DeclareAndDisplay

```
{  
    public static void main()  
    {  
        float x; // Declaring x of float type  
        float y; // Declaring y of float type  
        int m; // Declaring m of integer type  
        x = 75.86F;  
        y = 43.48F;  
        m = x + y; // This line will create an ERROR. Reason given below.  
        Console.WriteLine("m = x + y = 75.86 + 43.48 = " +m);  
    }  
}  
  
//***** Comment on the output *****  
  
//We declared 2 float type variables.  
  
//Added them  
  
//Saved the result in an Integer variable  
  
//Since the result of addition of 2 float numbers is a float only ...  
  
//We cannot save that value in an integer variable.  
  
//C# has strict check for data conversions taking place.  
  
//It does not automatically converts a larger data type to smaller one since it will create a loss of data.  
  
//For this purpose, we need to explicitly make the integer variable 'm' to float type.  
  
//If 'm' is also a float variable, then the output would have been like this ...  
  
//m = x + y = 75.86 + 43.48 = 119.34
```

4.3 – Program with a function

```
class ABC
{
    static int m;
    int n;
    void fun(int x, ref int y, out int z, int [] a)
    {
        int j = 10;
    }
}
```

```
// ***** Comment on Output *****
```

```
// The out parameter 'z' must be assigned to before the control leaves the current method
```

4.4 - Demonstrating Boxing & Unboxing

```
using System;

class Boxing
{
    public static void main(string[] a)
    {
        // ***** BOXING *****

        int m = 10;

        object om = m; // creates a box to hold m

        m = 20;

        Console.WriteLine("***** BOXING *****");

        Console.WriteLine("m = " + m); // m = 20

        Console.WriteLine("om = " +om); // om = 10

        Console.ReadLine();

        // ***** UNBOXING *****

        int n = 10;

        object on = n; // box n (creates a box to hold n)

        int x = (int)on; // unbox on back to an int

        Console.WriteLine("***** UNBOXING *****");

        Console.WriteLine("n = " + n); // n = 20

        Console.WriteLine("on = " +on); // on = 10

        Console.ReadLine();

    }
}
```

4.5 – Demonstrating addition of byte type variables

using System;

class addition

```
{  
    public static void Main()  
    {  
        byte b1;  
        byte b2;  
        int b3; // We are required to declare b3 as byte BUT its declared as int. The reason is given below.  
        b1 = 100;  
        b2 = 200;  
  
        // Normally this is the addition statement  
        //    b3 = b1 + b2;  
  
        // However it gives an error that cannot convert 'int' to 'byte'.  
  
        // When b2 & b3 are added, we get an integer value which cannot be stored in byte b1  
  
        // Thus we will declare b3 as integer type & explicitly convert b2 & b3 to int.  
  
        b3 = (int)b1 + (int)b2;  
  
        Console.WriteLine("b1 = " + b1);  
        Console.WriteLine("b2 = " + b2);  
        Console.WriteLine("b3 = " + b3);  
        Console.ReadLine();  
    }  
}
```

OUTPUT

b1 = 100

b2 = 200

b3 = 300

4.6 – Implementing some custom console output

```
using System;

class Demo
{
    public static void Main()
    {
        Console.WriteLine("Hello, \"Ram\"!");

        // Output ---> Hello, "Ram"!

        // Reason --> Due to the \" character, the characters Ram is in double quotes

        Console.WriteLine("*\n**\n***\n****\n");

        //Reason --> Due to the \n character, we get each set of * in a new line.

        Console.ReadLine();
    }
}
```

OUTPUT

Hello, "Ram" !

*

**

4.7 – Printing a home like figure in the console

using System;

class Home

```
{  
  
    public static void Main()  
    {  
        Console.WriteLine(" /\ ");  
        Console.WriteLine(" /  \ ");  
        Console.WriteLine(" /   \ ");  
        Console.WriteLine(" ----- ");  
        Console.WriteLine(" \"   \" ");  
        Console.WriteLine(" \"   \" ");  
        Console.WriteLine(" \"   \" ");  
        Console.WriteLine("\n\n This is My Home.");  
        Console.ReadLine();  
    }  
}
```

OUTPUT

```
 /\  
 /  \  
 /   \  
-----  
"   "  
"   "  
"   "
```

4.8 – Executing some console statements

```
using System;
```

```
class Demo
```

```
{
```

```
    public static void Main()
```

```
    {
```

```
        int m = 100;
```

```
        long n = 200;
```

```
        long l = m + n;
```

```
        Console.WriteLine("l = " + l);
```

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

```
        // No error in the program.
```

```
    }
```

```
}
```

OUTPUT

```
l = 300
```


Chapter # 5

Sr.no	Topic	Date	Sign
1.	Computation of Integer Values taken from console	30/1/2007	
2.	Computation of Float Values taken from console	30/1/2007	
3.	Average of 3 numbers	30/1/2007	
4.	Finding circumference & area of a circle	30/1/2007	
5.	Checking for validity of an expression	30/1/2007	
6.	Converting Rs. To Paisa	30/1/2007	
7.	Converting temp. from Fahrenheit to Celsius	30/1/2007	
8.	Determining salvage value of an item	30/1/2007	
9.	Reading & displaying the computed output of a real no.	30/1/2007	
10.	Evaluating distance travelled by a vehicle	30/1/2007	
11.	Finding the EOQ(Economic Order Quantity) & TBO(Time between Orders)	30/1/2007	
12.	Finding the frequencies for a range of different capacitance.	30/1/2007	

Chapter # 6

Sr.no	Topic	Date	Sign
1.	Adding odd & even nos from 0 – 20 & adding nos. divisible by 7 between 100 - 200	6/1/07	
2.	Finding a solution of linear equation	6/1/07	
3.	Computing marks of students	6/1/07	
4.	Selecting students on the basis of some given criteria on marks	6/1/07	
5.	Printing Floyd's triangle	6/1/07	
6.	Computing seasonal discount of a showroom	6/1/07	
7.	Reading 'x', Correspondingly Printing 'y'	6/1/07	

Chapter # 5

(Operators & Expressions)

5.1 # Computation of Integer Values taken from console

```
using System;

class integerdemo
{
    public static void Main()
    {

        string s1,s2;
        int a,b;

        Console.Write("Enter no 1 # "); // Display to enter no. 1
        s1 = Console.ReadLine (); // save the number in a string variable s1
        a = int.Parse (s1); // the string s1 is converted into int type variable

        Console.Write("Enter no 2 # "); //Display to enter no. 2
        s2 = Console.ReadLine (); // save the number in a string variable s2
        b = int.Parse (s2); // the string s2 is converted into int type variable

        // Here er converted both the string variables to int because we wanted to do
        // integer / numeric manipulation with the inputted string variables

        Console.WriteLine(""); // Blank line
        Console.WriteLine("***** Integer manipulations
*****");
        Console.WriteLine(""); // Blank line

        // Integer manipulations

        Console.WriteLine("No1 + No2 = " + (a+b));
        Console.WriteLine("No1 - No2 = " + (a-b));
        Console.WriteLine("No1 / No2 = " + (a/b));
        Console.WriteLine("No1 * No2 = " + (a*b));
        Console.WriteLine("No1 % No2 = " + (a%b));

        Console.ReadLine();

    }
}
```

Output:

Enter no 1 # 25

Enter no 2 # 15

***** Integer manipulations *****

No1 + No2 = 40

No1 - No2 = 10

No1 / No2 = 1

No1 * No2 = 375

No1 % No2 = 10

5.2 # Computation of Float Values taken from console

```
using System;
using System;

class floatdemo
{
    public static void Main()
    {
        string s1,s2;
        float a,b;

        Console.Write("Enter no 1 # "); // Display to enter no. 1
        s1 = Console.ReadLine (); // save the number in a string variable s1
        a = float.Parse (s1); // the string s1 is converted into float type variable

        Console.Write("Enter no 2 # "); //Display to enter no. 2
        s2 = Console.ReadLine (); // save the number in a string variable s2
        b = float.Parse (s2); // the string s2 is converted into float type variable

        // Here er converted both the string variables to float because we wanted to
do        // float / numeric manipulation with the inputted string variables

        Console.WriteLine(""); // Blank line
        Console.WriteLine("***** Integer manipulations
*****");
        Console.WriteLine(""); // Blank line

        // Integer manipulations

        Console.WriteLine("No1 + No2 = " + (a+b));
        Console.WriteLine("No1 - No2 = " + (a-b));
        Console.WriteLine("No1 / No2 = " + (a/b));
        Console.WriteLine("No1 * No2 = " + (a*b));
        Console.WriteLine("No1 % No2 = " + (a%b));

        Console.ReadLine();

    }
}
```

Output:

Enter no 1 # 25.64

Enter no 2 # 15.87

***** Float manipulations *****

No1 + No2 = 41.51

No1 - No2 = 9.77

No1 / No2 = 1.615627

No1 * No2 = 406.9068

No1 % No2 = 9.77

5.3 # Average of 3 numbers

```
using System;

class average
{
    public static void Main()
    {
        float a = 25;
        float b = 75;
        float c = 100;
        float avg = (a+b+c)/3;
        Console.WriteLine("The average of 25, 75 & 100 = " + avg);
        Console.ReadLine();
    }
}
```

Output:

The average of 25, 75 & 100 = 6.6666666

5.4 # Finding circumference & area of a circle

```
using System;

class circle
{
    public static void Main()
    {
        float radius = 12.5F;
        float circumference, area;
        float pi = 3.1487F;

        circumference = 2 * pi * radius;
        area = pi * radius * radius;

        Console.WriteLine("The Radius of the circle = " + radius);
        Console.WriteLine(""); //Blank Line
        Console.WriteLine("Its Circumference = " + circumference);
        Console.WriteLine("Its Area = " + area);
        Console.ReadLine();
    }
}
```

Output:

The Radius of the circle = 12.5

Its Circumference = 78.7175

Its area = 491.9844

5.5 # Checking for validity of an expression

```
using System;

class CheckExpression
{
    public static void Main()
    {
        int x,y,a,b;
        x - y = 100;
        // gives error
        //"The left-hand side of an assignment must be a variable, property or
indexer"

        x - (y = 100);
        // gives error
        //"Only assignment, call, increment, decrement, and new object expressions
        // can be used as a statement"

    }
}
```

5.6 # Converting Rs. To Paisa

```
using System;

class Money
{
    public static void Main()
    {
        float RsF;
        string s;
        Console.WriteLine("Enter the amount in Rs. : ");
        s = Console.ReadLine();
        RsF = float.Parse(s);
        Console.WriteLine("Amount in paise = " +(RsF*100));
        Console.ReadLine();
    }
}
```

Output:

Enter the amount in Rs. : 15

Amount in paise = 1500

5.7#Converting temp. from Fahrenheit to Celsius

```
using System;

class Temperature
{
    public static void Main()
    {
        float fahrenheit, celcius;
        string s;
        Console.Write("Enter the temperature in fahrenheit : ");
        s = Console.ReadLine();
        fahrenheit = float.Parse(s);
        celcius = (float)((fahrenheit-32)/1.8);
        Console.WriteLine("The Temperature in celcius = " +celcius);
        Console.ReadLine();
    }
}
```

Output:

Enter the temperature in fahrenheit : 98

Temperature in celcius = 36.66667

5.8 # Determining salvage value of an item

```
using System;

class depreciation
{
    public static void Main()
    {
        float depreciation, PurchasePrice, Yrs, SalvageValue;
        string d,p,y;
        // string variables are to store the values inputted in the console
        // each string variable has its character as that of the corresponding
        // starting character of float type variable

        Console.WriteLine("Enter the Depreciation : ");
        d = Console.ReadLine();
        depreciation = float.Parse(d);

        Console.WriteLine("Enter the PurchasePrice : ");
        p = Console.ReadLine();
        PurchasePrice = float.Parse(p);

        Console.WriteLine("Enter the Amount of Years : ");
        y = Console.ReadLine();
        Yrs = float.Parse(y);

        SalvageValue = (float)(PurchasePrice - (depreciation * Yrs));

        Console.WriteLine("SalvageValue = " + SalvageValue);

        Console.ReadLine();
    }
}
```

Output:

Enter the Depreciation : 50

Enter the PurchasePrice :15000

Enter the Amount of Years : 15

SalvageValue = 3456.4564

5.11 # Evaluating distance travelled by a vehicle

```
using System;

class Distance
{
    public static void Main()
    {
        float distance,u,t,a;
        string u1,t1,a1,reply;

        // u = Initial velocity
        // t = Time intervals
        // a = Acceleration
        // reply is the value used to check for again restart the program with
different values

        int replyforrestart,counter;
        // replyforrestart will take values either 0 or 1.
        // 1 means restart for next set of values, 0 means exit the program
        // counter is used for checking the no. of times the set of values occurs

        Console.WriteLine("***** This will calculate the distance travelled by a
vehicle *****");

        counter = 1;
        // For the first run, counter = 1

        startfromhere: // The program will restart from here for another set of
values.

        distance = u = t = a = 0.0F; //resetting all values to 0

        Console.WriteLine(""); // Blank Line

        Console.WriteLine("Set of value = " + counter);
        // Displays the no. of set of value

        Console.WriteLine(""); // Blank Line

        Console.Write("Enter the time interval (t) : ");
        t1 = Console.ReadLine();
        t = float.Parse(t1);

        Console.Write("Enter the initial velocity (u) : ");
        u1 = Console.ReadLine();
        u = float.Parse(u1);

        Console.Write("Enter the Acceleration (a) : ");
        a1 = Console.ReadLine();
        a = float.Parse(a1);

        distance = u*t + a*t*t/2;

        Console.WriteLine("Distance travelled by the vehicle = " + distance);

        Console.WriteLine(""); // Blank Line
```

```

        Console.WriteLine("Do you want to check for another values (1 for Yes / 0 to Exit)
? : ");

        reply = Console.ReadLine();

        replyforrestart = int.Parse(reply);

        if (replyforrestart == 1)
        {
            counter = counter+ 1;
            Console.WriteLine(""); // Blank Line
            Console.WriteLine("
*****
");
            goto startfromhere;
        }
        else
        {
            // Do nothing ... Simply program exits
        }
    }
}

```

Output:

***** This will calculate the distance travelled by a vehicle *****

Set of value = 1

Enter the time interval (t) : 15

Enter the initial velocity (u) : 10

Enter the Acceleration (a) : 150

Distance travelled by the vehicle = 17025

Do you want to check for another values (1 for Yes / 0 to Exit) ? : 1

Set of value = 2

Enter the time interval (t) : 25

Enter the initial velocity (u) : 5

Enter the Acceleration (a) : 540

Distance travelled by the vehicle = 168875

Do you want to check for another values (1 for Yes / 0 to Exit) ? : 0

5.11#Finding the EOQ(Economic Order Quantity) & TBO(Time between Orders)

```
using System;

class InventoryManagement
{
    public static void Main()
    {
        float dr,sc,cpu;
        //dr = Demand rate, sc = setup costs, cpu = cost per unit

        double EOQ,TBO;
        // EOQ = Economic Order Quaitity
        // TBQ = Optimal Time Between orders

        Console.WriteLine("\t\t\t\t\t ***** Inventory Management System *****");

        Console.WriteLine(""); // Blank Line

        Console.Write("Enter the Demand Rate : ");
        dr = float.Parse(Console.ReadLine());

        Console.Write("Enter the Setup Costs : ");
        sc = float.Parse(Console.ReadLine());

        Console.Write("Enter the Cost Per Unit : ");
        cpu = float.Parse(Console.ReadLine());

        Console.WriteLine(""); // Blank Line

        EOQ = Math.Sqrt(2*dr*sc/cpu); // Calculating EOQ

        TBO = Math.Sqrt(2*sc/(dr*cpu)); // Calculating TBO

        Console.WriteLine("Economic Order Quaitity = " +EOQ);
        Console.WriteLine("Optimal Time Between orders = " +TBO);

        Console.ReadLine();
    }
}
```

Output:

Enter the Demand Rate : 150

Enter the Setup Costs : 250

Enter the Cost Per Unit : 25

Economic Order Quaitity = 54.772255

Optimal Time Between orders = 0.3654837167

5.12 # Finding the frequencies for a range of different capacitance.

```
using System;

class ElectricalCircuit
{
    public static void Main()
    {
        float L,R,C,Frequency;

        // L = Inductance
        // R = Resistance
        // C = Capacitance

        //double Frequency;

        Console.WriteLine(" ***** Calculating frequencies for different values of
Capacitance *****");

        Console.WriteLine(""); // Blank Line

        Console.Write("Enter the Inductance (L) : ");
        L = float.Parse(Console.ReadLine());

        Console.Write("Enter the Resistance (R) : ");
        R = float.Parse(Console.ReadLine());

        Console.WriteLine(""); // Blank Line

        for (C = 0.01F; C <= 0.1; C = C + 0.01F)
        {

            Frequency = (float)(Math.Sqrt((1/L*C)-((R*R)/(4*C*C))));
            Console.WriteLine("For Capacitance " + C + ", The Frequency = " +
Frequency);
        }

        Console.ReadLine();
    }
}
```

Output :

***** Calculating frequencies for different values of Capacitance *****

Enter the Inductance (L) : 0.00004

Enter the Resistance (R) : 0.00008

For Capacitance 0.01, The Frequency = 15.81139

For Capacitance 0.02, The Frequency = 22.36068

For Capacitance 0.03, The Frequency = 27.38613

For Capacitance 0.04, The Frequency = 31.62278

For Capacitance 0.05, The Frequency = 35.35534

For Capacitance 0.06, The Frequency = 38.72983

For Capacitance 0.07, The Frequency = 41.833

For Capacitance 0.08, The Frequency = 44.72136

For Capacitance 0.09, The Frequency = 47.43416

For Capacitance 0.1, The Frequency = 50

Chp = 6

(Decision Making & Branching)

6.1 # Adding odd & even nos from 0 – 20 & adding nos. divisible by 7 between 100 – 200

```
using System;

class SumOfOdds
{
    public static void Main()
    {
        int x=0, sumodd=0, sumeven=0, sumdiv7 = 0 ,totalno7 = 0, i;
        // here ...
        // "sumodd" will contain sum of all odd the numbers from 1 - 20
        // "sumeven" will contain sum of all even the numbers from 1 - 20
        // "sumdiv7" will contain the sum of all numbers from 100 - 200 divisible by 7
        // "totalno7" will contain the total no. of all numbers from 100 - 200
        // "i" is a variable used in loops
        // "x" is a temporary variable which check for the conditions imposed on it

        // checking for the odd & even numbers
        for (i=0 ; i<=20 ; i++)
        {
            x = i % 2;
            if (x != 0)
            {
                sumodd = sumodd + i;
            }
            if (x == 0)
            {
                sumeven = sumeven + i;
            }
        }

        //checking for the sum & no. of numbers divisible by 7

        x = 0; // resetting the value of 'x'
        for (i=100; i<=200;i++)
        {
            x = i % 7;
            if (x == 0)
            {
                sumdiv7 = sumdiv7 + i;
                totalno7 = totalno7 + 1;
            }
        }

        Console.WriteLine("Sum of all odd numbers from 1 - 20 = " + sumodd + "\n");
        Console.WriteLine("Sum of all even numbers from 1 - 20 = " + sumeven + "\n");
        Console.WriteLine("Sum of all numbers from 100 - 200, divisible by 7 = " +
sumdiv7 + "\n");
        Console.WriteLine("Total numbers from 100 - 200, divisible by 7 = " +
totalno7 + "\n");

        Console.ReadLine();
    }
}
```

Output:

Sum of all odd numbers from 1 - 20 = 100

Sum of all even numbers from 1 - 20 = 110

Sum of all numbers from 100 - 200, divisible by 7 = 2107

Total numbers from 100 - 200, divisible by 7 = 14

6.2 # Finding Solution of linear equations

```
using System;

class LinearEquations
{
    public static void Main()
    {
        int response;
        float a,b,c,d,m,n, temp;
        double x1,x2;

        EnterNewValuesAgain:

        Console.WriteLine(""); // Blank Line
        Console.WriteLine("***** Linear Equation
*****");
        Console.WriteLine(""); // Blank Line
        // Reading the value of a
        Console.Write("Enter the value of a : ");
        a = float.Parse(Console.ReadLine());

        // Reading the value of b
        Console.Write("Enter the value of b : ");
        b = float.Parse(Console.ReadLine());

        // Reading the value of c
        Console.Write("Enter the value of c : ");
        c = float.Parse(Console.ReadLine());

        // Reading the value of d
        Console.Write("Enter the value of d : ");
        d = float.Parse(Console.ReadLine());

        temp = a*d - b*c;

        if (temp == 0)
        {
            Console.WriteLine(""); // Blank Line
            Console.WriteLine("The denominator equals to zero (0); Cannot proceed
further ...");
            Console.Write("Do You want to enter new values (1 For Yes / 0 For No) ?
");
            response = int.Parse(Console.ReadLine());

            if (response == 0)
            {
                goto Exit;
            }
            else
            {
                goto EnterNewValuesAgain;
            }
        }
        else
        {
            // Reading the value of m
            Console.Write("Enter the value of m : ");
            m = float.Parse(Console.ReadLine());
        }
    }
}
```

```

// Reading the value of n
Console.Write("Enter the value of n : ");
n = float.Parse(Console.ReadLine());

x1 = ((m*d) + (b*n))/((a*d) - (c*b));
x2 = ((n*a) + (m*c))/((a*d) - (c*b));

Console.WriteLine(""); // Blank Line

Console.WriteLine("Value of x1 = " + x1);
Console.WriteLine("Value of x2 = " + x2);

Console.WriteLine(""); // Blank Line

Console.Write("Do You want to enter new values (1 For Yes / 0 For No) ?
");

response = int.Parse(Console.ReadLine());
if (response == 0)
{
    goto Exit;
}
else
{
    goto EnterNewValuesAgain;
}

}

Exit:
    Console.WriteLine(""); // Blank Line
    Console.WriteLine("Thank You For using this small program ... :)");
    Console.ReadLine();
}
}

```

Output:

***** Linear Equation *****

Enter the value of a : 5

Enter the value of b : 5

Enter the value of c : 5

Enter the value of d : 5

The denominator equals to zero (0); Cannot proceed further ...

Do You want to enter new values (1 For Yes / 0 For No) ? 1

***** Linear Equation *****

Enter the value of a : 15

Enter the value of b : 5

Enter the value of c : 3

Enter the value of d : 20

Enter the value of m : 5

Enter the value of n : 6

Value of x1 = 0.4561

Value of x2 = 0.364821

Do You want to enter new values (1 For Yes / 0 For No) ? 0

You For using this small program ... :)

6.5 # Computing marks of students

```
using System;

class MarksRange
{
    public static void Main()
    {
        int i, count80 = 0, count60 = 0, count40 = 0, count0 = 0;
        float [] marks =
{57.5F,45.9F,98.01F,56.4F,46.5F,80,82,67,76,49,91,55,78,79,19.5F,25.8F,35,36,35,28,25.8F,4
6,55,59,68,97,85,48.5F,67,84};

        for (i = 0; i<=29; i++)
        {
            if(marks[i] > 80 && marks [i] < 101)
            {
                count80 = count80 + 1;
            }
            else if(marks [i] > 60 && marks[i] < 81)
            {
                count60 = count60 + 1;
            }
            else if(marks [i] > 40 && marks[i] < 61)
            {
                count40 = count40 + 1;
            }
            else
            {
                count0 = count0 + 1;
            }
        }

        Console.WriteLine("Students in the range of 81 - 100 : "+ count80);
        Console.WriteLine("Students in the range of 61 - 80 : "+ count60);
        Console.WriteLine("Students in the range of 41 - 60 : "+ count40);
        Console.WriteLine("Students in the range of 0 - 40 : "+ count0);

        Console.ReadLine();
    }
}
```

Output:

```
Students in the range of 81 - 100 : 6
Students in the range of 61 - 80 : 7
Students in the range of 41 - 60 : 10
Students in the range of 0 - 40 : 7
```

6.7 # Selecting students on the basis of some given criteria on marks

```
using System;

class Admission
{
    public static void Main()
    {
        float mksMaths, mksPhysics, mksChemistry, mksTotal, MathsPhysics;
        int response;

        beginning:

        Console.WriteLine(""); // Blank Line

        Console.WriteLine("          ***** Students Enrollment Checking Criteria
***** ");

        Console.WriteLine(""); // Blank Line

        Console.Write("Enter the marks in Maths : ");
        mksMaths = float.Parse(Console.ReadLine());

        Console.Write("Enter the marks in Chemistry : ");
        mksChemistry = float.Parse(Console.ReadLine());

        Console.Write("Enter the marks in Physics : ");
        mksPhysics = float.Parse(Console.ReadLine());

        mksTotal = (float)(mksMaths + mksChemistry + mksPhysics);

        MathsPhysics = (float)(mksMaths + mksPhysics);

        if ((mksMaths >= 60 && mksPhysics >= 50 && mksChemistry >= 40) || (mksTotal >=
200 || (mksMaths + mksPhysics) >= 150))
        {
            Console.WriteLine("Congratulations !!! The candidate is selected ... ");
        }
        else
        {
            Console.WriteLine("Sorry, The candidate is rejected ... Better luck for
next year.");
        }

        Console.WriteLine(""); // Blank Line

        Console.Write("Enter 1 for next candidate, 0 to exit : ");
        response = int.Parse(Console.ReadLine());

        if (response == 1)
            goto beginning;
        else
            goto end;

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

Output:

***** Students Enrollment Checking Criteria *****

Enter the marks in Maths : 50

Enter the marks in Chemistry : 40

Enter the marks in Physics : 35

Sorry, The candidate is rejected ... Better luck for next year.

Enter 1 for next candidate, 0 to exit : 1

***** Students Enrollment Checking Criteria *****

Enter the marks in Maths : 70

Enter the marks in Chemistry : 80

Enter the marks in Physics : 85

Congratulations !!! The candidate is selected ...

Enter 1 for next candidate, 0 to exit : 0

6.8 # Floyd's Triangle

```
using System;

class FloydTriangle1
{
    public static void Main()
    {
        int i,j,k=1;
        Console.WriteLine("          ***** Floyd's Triangle - Normal Numeric Mode
***** ");

        for (i=1; i<=13; i++)
        { // 13 is the height of the triangle
            for (j=1; j<i+1; j++)
            { // each time the number per line is incremented by 1
                Console.Write(k++ + " "); // k is the actual data (number) which
will be printed.
            }
            Console.Write("\n"); // then we go to the next line.
        }

        Console.ReadLine();
    }
}
```

Output:

```
          ***** Floyd's Triangle - Normal Numeric Mode *****

1
2 3
4 5 6
7 8 9 10
11 12 13 14 15
16 17 18 19 20 21
22 23 24 25 26 27 28
29 30 31 32 33 34 35 36
37 38 39 40 41 42 43 44 45
46 47 48 49 50 51 52 53 54 55
56 57 58 59 60 61 62 63 64 65 66
67 68 69 70 71 72 73 74 75 76 77 78
79 80 81 82 83 84 85 86 87 88 89 90 91
```

6.9 # Computing seasonal discount of a showroom

```
using System;

class SeasonalDiscount
{
    public static void Main()
    {
        int amt;
        float Mill_disc, Hand_disc, DiscountedAmt;

        Console.WriteLine("***** Seasonal Discount of a Mall
***** ");

        Console.Write("Enter the Purchase amount : ");
        amt = int.Parse(Console.ReadLine());

        if (amt >= 0 && amt <= 100)
        {
            Mill_disc = amt * 0.0F;
            Hand_disc = amt * 0.05F;

            Console.WriteLine("\n\n You Made a purchase of : " + amt + "Rs.");
            Console.WriteLine("\nYou are eligible to recieve a discount of : \n" +
(amt - Mill_disc) + " Rs. = (0%) on Mill Cloth & \n" + (amt - Hand_disc) + "Rs. = (5%) on
HandLoom Items." );
            DiscountedAmt = amt - (Mill_disc + Hand_disc);
            Console.WriteLine("\nAfter all the discounts, you need to pay a sum of "
+ DiscountedAmt + " instead of " + amt + ", \nthus making a Profit of " + (Mill_disc +
Hand_disc) + "Rs.");
        }

        else if (amt >= 101 && amt <= 200)
        {
            Mill_disc = amt * 0.05F;
            Hand_disc = amt * 0.75F;

            Console.WriteLine("\n\n You Made a purchase of : " + amt + "Rs.");
            Console.WriteLine("\nYou are eligible to recieve a discount of : \n" +
(amt - Mill_disc) + " Rs. = (5%) on Mill Cloth & \n" + (amt - Hand_disc) + "Rs. = (7.5%)
on HandLoom Items." );
            DiscountedAmt = amt - (Mill_disc + Hand_disc);
            Console.WriteLine("\nAfter all the discounts, you need to pay a sum of "
+ DiscountedAmt + " instead of " + amt + ", \nthus making a Profit of " + (Mill_disc +
Hand_disc) + "Rs.");
        }

        else if (amt >= 201 && amt <= 300)
        {
            Mill_disc = amt * 0.75F;
            Hand_disc = amt * 0.1F;

            Console.WriteLine("\n\n You Made a purchase of : " + amt + "Rs.");
            Console.WriteLine("\nYou are eligible to recieve a discount of : \n" +
(amt - Mill_disc) + " Rs. = (7.5%) on Mill Cloth & \n" + (amt - Hand_disc) + "Rs. = (10%)
on HandLoom Items." );
            DiscountedAmt = amt - (Mill_disc + Hand_disc);
            Console.WriteLine("\nAfter all the discounts, you need to pay a sum of "
+ DiscountedAmt + " instead of " + amt + ", \nthus making a Profit of " + (Mill_disc +
Hand_disc) + "Rs.");
        }
    }
}
```

```

else if (amt > 300)
{
    Mill_disc = amt * 0.1F;
    Hand_disc = amt * 0.15F;

    Console.WriteLine("\n\n You Made a purchase of : " +amt + "Rs.");
    Console.WriteLine("\nYou are eligible to recieve a discount of : \n" +
(amt - Mill_disc) + " Rs. = (10%) on Mill Cloth & \n" + (amt - Hand_disc) + "Rs. = (15%)
on HandLoom Items." );
    DiscountedAmt = amt - (Mill_disc + Hand_disc);
    Console.WriteLine("\nAfter all the discounts, you need to pay a sum of "
+ DiscountedAmt + " instead of " + amt + ", \nthus making a Profit of " + (Mill_disc +
Hand_disc) + "Rs.");

}

Console.ReadLine();
}
}

```

Output:

***** Seasonal Discount of a Mall *****

Enter the Purchase amount : 250

You Made a purchase of : Rs. 100

You are eligible to receive a discount of :

0 Rs. (0%) on Mill Items

5Rs. (5 %) on Handloom Items

nAfter all the discounts, you need to pay a sum of 95, instead of 100, thus making a profit of 5Rs.

6.10 # Reading 'x', Correspondingly Printing 'y'

```
using System;

class ChangingValuesOfY
{
    public static void Main()
    {
        int x,y;

        Console.WriteLine("Enter the value of x : ");
        x = int.Parse(Console.ReadLine());

        Console.WriteLine(""); // Blank Line
        Console.WriteLine(""); // Blank Line

        Console.WriteLine("          ***** Changing values of Y by nested if
statements *****");

        Console.WriteLine(""); // Blank Line

        if (x != 0)
        {
            if (x > 0)
            {
                Console.WriteLine("Y = 1");
            }

            if (x < 0)
            {
                Console.WriteLine("Y = -1");
            }
        }
        else
        {
            Console.WriteLine("Y = 0");
        }

        Console.WriteLine(""); // Blank Line

        Console.WriteLine("          ***** Changing values of Y by else if
statements *****");

        Console.WriteLine(""); // Blank Line

        if (x == 0)
        {
            Console.WriteLine("Y = 0");
        }
        else if(x > 0)
        {
            Console.WriteLine("Y = 1");
        }
        else
        {
            Console.WriteLine("Y = -1");
        }
    }
}
```

```

        Console.WriteLine(""); // Blank Line

        Console.WriteLine("          ***** Changing values of Y by conditional
operator *****");

        Console.WriteLine(""); // Blank Line

        y = (x != 0)?((x>0)?1:-1):0;

        Console.WriteLine("Y = "+y);

        Console.ReadLine();

    }
}

```

Output:

Enter the value of x : 5

```

          ***** Changing values of Y by nested if statements *****

Y = 1

          ***** Changing values of Y by else if statements *****

Y = 1

          ***** Changing values of Y by conditional operator *****

Y = 1

```

CHAPTER # 7

Sr.no	Topic	Date	Sign
1.	Reversing the numbers	13/02/2007	
2.	Finding the factorial of a given number	13/02/2007	
3.	Calculating the sum of digits of the given number	13/02/2007	
4.	Printing & adding Fibonacci series	13/02/2007	
5.	Investment Equation	13/02/2007	
6.	Converting \$ into Rs.	13/02/2007	
7.	Demonstrating use of break, continue & goto	13/02/2007	

INDEX FOR CHP 8, 9 & 10

Sr.no	Topic	Date	Sign
1.	Printing triangles into various formats	27/2/07	
2.	Calculate standard deviation & mean of the array elements	27/2/07	
3.	Finding the maximum & minimum of 3 numbers entered	27/2/07	
4.	Finding largest array element & average of array elements via methods	27/2/07	
5.	Sorting 2 arrays & merging into 1	27/2/07	
6.	Accepting a list of 5 items	27/2/07	
7.	Counting number of words in a string	27/2/07	
8.	Reversing array by creating a method 'Reverse'	27/2/07	
9.	Read an array & sort it	27/2/07	

7.1 # Reversing the numbers

```
using System;

class ReverseNumber
{
    public static void Main()
    {
        int num,rem,i,counter=0,temp;
        // num : Contains the actual number inputted via the console
        // rem : remainder of the number 'num' when divided by 10
        // i : loop variable
        // counter : determines the no. of digits in the inputted number 'num'
        // temp : temporary variable used to save the value of 'num' (Explained
further)

        Console.WriteLine("Enter an integer number (Not more than 9 digits) : ");
        num = int.Parse(Console.ReadLine());

        temp = num;
        // Here we are saving 'num' in 'temp' coz its value after determining the no.
of digits will loose.
        // So after its work is done, 'num' will contain value = 0
        // The value of 'num' is resetted to its original value later from 'temp'
variable

        // ***** Determine the no. of digits in the inputted number
*****
        while(num > 0)
        {
            rem = num % 10;
            num = num / 10;
            if (num <= 0)
            {
                break;
            }
            else
            {
                counter = counter + 1;
            }
        }

        Console.WriteLine("Number of digits are = " + (counter+1));

        Console.WriteLine("The reversed digits are : ");

        rem = 0;
        // resetting the value of remainder 'rem'

        num = temp;
        // resetting the lost value of 'num' from 'temp'

        // ***** Determine the reversed of inputted digits
*****
        // Funda :
        // 1) Divide the number by 10 & determine the remainder. (Save the remainder
in 'rem')
```

```
// This will give us the last digit in the actual inputted number.
//
// 2) Write the number so obtained into the console
//
// 3) Divide the same number by 10 & get the quotient this time.
// Since division is between the integers, we will get the new number,
deprived of the last digit.
// Then again goto step 1) & continue until & unless the counter is equal to
'i' (coz thats the loop varibale)

    for(i = 0; i<=counter; i++)
    {
        rem = num % 10;
        Console.Write(rem);
        num = num / 10;
    }

    Console.ReadLine();

}
}
```

Output:

Enter an integer number (Not more than 9 digits) : 3547786

Number of digits are = 7

The reversed digits are : 6877453

7.2 # Finding the factorial of a given number

```
using System;

class Factorial
{
    public static void Main()
    {
        int no,i,fact=1;
        Console.Write("Enter a number to find its factorial : ");
        no = int.Parse(Console.ReadLine());

        if (no != 0)
        {
            for (i = no; i>=1; i--)
            {
                fact = fact * i;
            }
            Console.WriteLine("Factorial = " +fact);
        }
        else
        {
            Console.WriteLine("You entered 0, not valid.");
        }

        Console.ReadLine();
    }
}
```

Output:

Enter a number to find its factorial : 9

Factorial = 362880

7.3 #Calculating the sum of digits of the given number

```
using System;

class SumOfNumbers
{
    public static void Main()
    {
        int num,rem,i,counter=0,temp,sum=0;
        // num : Contains the actual number inputted via the console
        // rem : remainder of the number 'num' when divided by 10
        // i : loop variable
        // counter : determines the no. of digits in the inputted number 'num'
        // temp : temporary variable used to save the value of 'num' (Explained
further)

        Console.WriteLine("Enter an integer number (Not more than 9 digits) : ");
        num = int.Parse(Console.ReadLine());

        temp = num;
        // Here we are saving 'num' in 'temp' coz its value after determining the no.
of digits will loose.
        // So after its work is done, 'num' will contain value = 0
        // The value of 'num' is resetted to its original value later from 'temp'
variable

        // ***** Determine the no. of digits in the inputted number
*****
        while(num > 0)
        {
            rem = num % 10;
            num = num / 10;
            if (num <= 0)
            {
                break;
            }
            else
            {
                counter = counter + 1;
            }
        }

        Console.WriteLine("Number of digits are = " + (counter+1));

        rem = 0;
        // resetting the value of remainder 'rem'

        num = temp;
        // resetting the lost value of 'num' from 'temp'

        // ***** Determine the reversed of inputted digits
*****
        // Funda :
        // 1) Divide the number by 10 & determine the remainder. (Save the remainder
in 'rem')
        // This will give us the last digit in the actual inputted number. (Same as
reversing numbers logic)
        //
```

```
        // 2) Add the number so obtained into the variable 'sum'
        //
        // 3) Divide the same number by 10 & get the quotient this time.
        // Since division is between the integers, we will get the new number,
deprived of the last digit.
        // Then again goto step 1) & continue until & unless the counter is equal to
'i' (coz thats the loop varibale)

        for(i = 0; i<=counter; i++)
        {
            rem = num % 10;
            sum = sum + rem;
            num = num / 10;
        }
        Console.WriteLine("Sum = " +sum);
        Console.ReadLine();
    }
}
```

Output:

Enter an integer number (Not more than 9 digits) : 65478457

Number of digits : 8

Sum of digits : 46

7.4 # Printing & adding Fibonacci series

```
using System;

class Fibonacci
{
    public static void Main()
    {
        int first = 1, second = 1, third, no, count = 0;
        long sum = 2;
        // 'first', 'second', 'third' are the first, second & third numbers in the
        fibonacci series
        // 'first' & 'second' are both initialised to 1
        // sum of 'first' & 'second' are added to the 'third' variable
        // 'sum' will contain the sum of all the digits in the fibonacci series. It is
        initialies to 2 coz sum of first 2 digits is 2
        // 'no' is the number inputted from the console up till which the fibonacci
        series is displayed
        // 'count' counts the number of digits in the fibonacci series

        Console.WriteLine("Enter the number uptill which you want the fibonacci numbers :
");
        no = int.Parse(Console.ReadLine());

        if (no >= 45)
        {
            // checking for values out of range.
            Console.WriteLine("Out of range values. Dont enter more than 45.");
            goto exit;
        }

        Console.WriteLine("Fibonacci Series : 1 1");
        // Initial 2 numbers of the fibonacci series are just '1' & '1', thus writing
        it directly

        do
        {
            third = first + second;
            // adding 'third' = 'first' + 'second'

            Console.WriteLine(" "+third);
            // display the 'third' digit in the series

            first = second;
            // make 'first' digit, the 'second' one
            second = third;
            // make 'second' digit, the 'third' one

            // we did this coz in fibonacci series, each digit is a sum of previous
            2 digits

            count = count + 1;
            // increment the counter

            sum = sum + third;
            // add the sum in the 'sum' variable from 'third' variable

        }
        while((count + 3) <= no);
        // we entered the 'no' from the console & also the first 2 digits are not from
        this loop
    }
}
```

```
        // thus we added +3 here to the 'count' variable so that we get the exact
specified no. of digits.
        // if we didnt added 3, then the series will go beyond the specified number of
digits from the console via 'no'

        Console.WriteLine("\nSum of all fibonacci digits : " +sum);
        // Display the sum
        exit:
        Console.ReadLine();
    }
}
```

Output:

Enter the number uptill which you want the fibonacci numbers : 8

Fibonacci Series : 1 1 2 3 5 8 13 21 34

Sum of all Fibonacci digits : 88

7.5 #Investment Equation

```
using System;

class Investment
{
    public static void Main()
    {
        int P=1000,n;
        float r=0.1F;
        double V;

        Console.WriteLine("***** Investement Option of 10 yrs
*****");

        Console.WriteLine(""); // Blank Line

        Console.WriteLine("Principal(P)    Rate(r)    Number Of Yrs(n)    Value
Of Money(V)\n");
        Console.WriteLine("-----\n");
        V = P * (1 + r);
        for (n=1;n<=10;n++)
        {
            Console.WriteLine ("    " + P + "    " + r + "
" + n + "    " + V);
            P = P + 1000;
            r = r + 0.01F;
            V = P * (1 + r);
        }

        Console.ReadLine();
    }
}
```

Output:

```
***** Investement Option of 10 yrs *****
Principal(P)    Rate(r)    Number Of Yrs(n)    Value Of Money(U)
-----
1000            0.1       1                    1100.00000149012
2000            0.11      2                    2219.99999880791
3000            0.12      3                    3359.99999195337
4000            0.13      4                    4519.99998092651
5000            0.14      5                    5700.00000298023
6000            0.15      6                    6900.00003576279
7000            0.16      7                    8120.00007927418
8000            0.17      8                    9360.0001335144
9000            0.18      9                    10620.0001984835
10000           0.19     10                   11900.0002741814
```

7.7 # Converting \$ into Rs.

```
using System;

class DollarToRupees
{
    public static void Main()
    {
        float dol,rs,current;
        int i;

        Console.WriteLine("What is the current value of 1 $ as per INDIAN Rs. : ");
        current = float.Parse(Console.ReadLine());

        Console.WriteLine(""); // Blank Line

        for (i=1;i<=5;i++)
        {
            Console.WriteLine("Enter value " + i + " in Dollars : ");
            dol = float.Parse(Console.ReadLine());
            rs = dol * current;
            Console.WriteLine(dol + " $ = " +rs + "Rs.");
            Console.WriteLine(""); // Blank Line
        }

        Console.ReadLine();
    }
}
```

Output:

What is the current value of 1 \$ as per INDIAN Rs. : 48.5

Enter value 1 in Dollars : 50
50 \$ = 2425Rs.

Enter value 2 in Dollars : 57.47
57.47 \$ = 2787.295Rs.

Enter value 3 in Dollars : 20
20 \$ = 970Rs.

Enter value 4 in Dollars : 3
3 \$ = 145.5Rs.

Enter value 5 in Dollars : 48.5
48.5 \$ = 2352.25Rs.

7.10 #Demonstrating use of break, continue & goto

```
using System;

class BreakContiuneGoto
{
    public static void Main()
    {
        int n = 10;
        while(n<200)
        {
            if(n<100)
            {
                if(n<50)
                {
                    goto lessthan50;
                }
                Console.Write(" " +n);
                n = n + 20;
                continue;
            }

            lessthan50:
            {
                Console.Write(" " +n);
                n = n + 10;
                continue;
            }

            if(n==50)
            {
                Console.WriteLine("");
                n = n + 10;
                continue;
            }

            if(n > 90)
                break;
            Console.Write(" " +n);
            n = n + 10;
        }
        Console.WriteLine();
        Console.ReadLine();
    }
}
```

Output:

10 20 30 40 50 60 70 80 90 110 120 130 140 150 160 170 180 190

7.6 - PRINTING TRIANGLES INTO VARIOUS FORMATS

a)

```
using System;

class DollarDesign
{
    public static void Main()
    {
        int no=1,i,j;

        for(i = 1 ; i < 6 ; i ++) // Outer loop for incrementing the numbers to be
displayed
        {
            Console.WriteLine(" "); // Leave a line after each new number
            for(j = 1; j < 6; j ++) // Inner loop to specify the numer of times the
particular number is to be printed.
            {
                Console.Write(no);
                if(i == j)
                    // If a number is printed that many number of times ...
                    // e.g. If 3 is there. The if 3 is printed 3 times, then
this condition arises
                {
                    no = no + 1; // Increment the number
                    goto loop1; // Goto outer loop
                }
            }
            loop1:continue;
        }
        Console.ReadLine();
    }
}
```

Output:

```
1
22
333
4444
55555
```

b)

```
using System;

class TriangleDollar
{
    public static void Main()
    {
        int i,j,k;
        string d="$";
        for(i=1;i<=5;i++)
        {
            for(k=1;k<=i;k++)
                Console.Write(" ");
            for(j=5;j>=i;j--)
            {
                Console.Write ("$,+j); // Enter the space with a '$' sign
// This is another syntax of Console.Write method. Here the digit after the comma `,'
signifies the position of the first character '$' on the output screen.
            }

            Console.WriteLine("\n"); // then we go to the next line.
        }
        Console.ReadLine();
    }
}
```

Output:

```
$$$$$
$$$$
$$$
$$
$
```

c)

```
using System;
class PyramidNumbers
{
    public static void Main()
    {
        int i,j,num=5,k;
        for(i=1;i<=num;i++)
        {
            for(k=num;k>=i;k--)// Loop for the blank spaces
            {
                Console.Write(" ");
            }

            for(j=1;j<=i;j++)// Loop for determining the number of times the number
is to be written
            {
                Console.Write(" " +i); // " " is a space needed in between the
numbers
            }

            Console.WriteLine("\n"); // Go to the next line for next number
        }
        Console.ReadLine();
    }
}
```

Output:

```
    1
   2 2
  3 3 3
 4 4 4 4
5 5 5 5 5
```

§.6 – CALCULATE STANDARD DEVIATION & MEAN OF THE ARRAY ELEMENTS

```
using System;

class StdDeviation
{
    public static void Main()
    {
        float [] nos = {3.5F,57,2,6,24,14,95,23,74,23};
        int n = nos.Length;
        float sum = 0.0F,sumofsq = 0.0F, mean;
        double deviation;

        Console.WriteLine("Array List consists of : ");

        for (int i = 0; i < n; i ++)
        {
            Console.WriteLine(nos[i] + " ");
        }

        for (int i = 0; i < n; i ++)
        {
            sum = sum + nos[i];
        }

        for (int i = 0; i < n; i ++)
        {
            sumofsq = sumofsq + (nos[i]*nos[i]);
        }

        mean = sum / n;
        deviation = Math.Sqrt(sumofsq / 8.0);

        Console.WriteLine("\n\n Sum = " +sum);
        Console.WriteLine("\n Mean = " +mean);
        Console.WriteLine("\n Deviation = " +deviation );
        Console.ReadLine();
    }
}
```

Output:

```
Array List consists of : 3.5 57 2 6 24 14 95 23 74 23
Sum = 321.5
Mean = 32.15
Deviation = 49.5381797202
```

§.13 & §.14 - FINDING THE MAXIMUM & MINIMUM OF 3 NUMBERS ENTERED

```
using System;
class LargestSmallest
{
    public static void Main()
    {
        int a,b,c,largest,smallest;

        Console.WriteLine("Enter No 1 : ");
        a = int.Parse(Console.ReadLine());

        Console.WriteLine("Enter No 2 : ");
        b = int.Parse(Console.ReadLine());

        Console.WriteLine("Enter No 3 : ");
        c = int.Parse(Console.ReadLine());

        if (a > b)
        {
            if(a > c)
            {
                largest = a;
            }
            else
            {
                largest = c;
            }
        }
        else
        {
            if(c>b)
            {
                largest = c;
            }
            else
            {
                largest = b;
            }
        }

        if (a < b)
        {
            if(a < c)
            {
                smallest = a;
            }
            else
            {
                smallest = c;
            }
        }
        else
        {
            if(c<b)
            {
                smallest = c;
            }
            else
            {
                smallest = b;
            }
        }
    }
}
```

```
    }  
  
    Console.WriteLine("\n\n The Largest Number = " +largest);  
    Console.WriteLine("\n The Smallest Number = " +smallest);  
    Console.ReadLine();  
} }  
}
```

Output:

```
Enter No 1 : 15  
Enter No 2 : 54  
Enter No 3 : 21
```

```
The Largest Number = 54  
The Smallest Number = 15
```

§.15 – FINDING LARGEST ARRAY ELEMENT & AVERAGE OF ARRAY ELEMENTS VIA METHODS.

```
using System;

class ArrayFunction
{
    public static void Main()
    {
        long Largest;
        double Average;
        int c;
        int num;
        int[] array1;

        Console.WriteLine("Enter the number of Elements in an Array : ");

        c=int.Parse(Console.ReadLine());

        array1=new int[c];

        for (int i=0 ; i<c ;i++)
        {
            Console.WriteLine("Enter the element " + i);
            num=int.Parse(Console.ReadLine());
            array1[i]=num;
        }

        foreach (int i in array1)
        {
            Console.Write(" " + i);
        }

        Console.WriteLine ();

        Largest = Large(array1);
        Average = Avg(array1);

        Console.WriteLine ("\n The largest element in the array is " +
            Largest);
        Console.WriteLine ("The Average of elements in the array is " +
            Average);
        Console.ReadLine();
    }

    // Determining the largest array element
    static int Large (params int [] arr)
    {
        int temp=0;

        for ( int i = 0; i < arr.Length; i++)
        {
            if (temp <= arr[i])
            {
                temp = arr[i];
            }
        }
        return(temp);
    }
}
```

```
// Determining the average of array elements
static double Avg (params int [] arr)
{
    double sum=0;

    for ( int i = 0; i < arr.Length; i++)
    {
        sum = sum + arr[i];
    }
    sum = sum/arr.Length;

    return(sum);
}
}
```

Output:

Enter the number of Elements in an Array : 5

Enter the element 1 : 5

Enter the element 2 : 7

Enter the element 3 : 3

Enter the element 4 : 1

Enter the element 5 : 8

largest element in the array is 8

The Average of elements in the array is 4.8

9.7 – SORTING 2 ARRAYS & MERGING INTO 1

```
using System;
class SortArray
{
    public static void Main()
    {
        int [] A={127,157,240,550,510};
        int [] B={275,157,750,255,150};
        int CLength=(A.Length +B.Length);
        int [] C=new int[CLength];
        int i=0,j=0,k;

        Console.WriteLine ("Sorted array list : ");
        for(k=0;k<=(i+j);k++)
        {
            if(A[i]<=B[j])
            {
                C[k]=A[i];
                Console.Write (C[k] + " ");
                if(i<4)
                {
                    i++;
                }
            }
            else
            {
                C[k]=B[j];
                Console.Write (C[k] + " ");
                if(j<4)
                {
                    j++;
                }
            }
        }
        for(i=0;i<CLength;i++)
        {
            Console.Write(C[i] + " ");
        }
        Console.ReadLine();
    }
}
```

Output :

Sorted array list : 127 150 157 157 240 255 275 510 550 750

9.11 - ACCEPTING A LIST OF 5 ITEMS

```
using System;
using System.Collections;

class ShoppingList
{
    public static void Main(string []args)
    {
        ArrayList n = new ArrayList ();
        n.Add(args[0]);
        n.Add(args[1]);
        n.Add(args[2]);
        n.Add(args[3]);
        n.Add(args[4]);

        n.Sort();
        Console.WriteLine ("The items in the Shopping List are : ");

        for (int i =0; i< nCount; i++)
        {
            Console.WriteLine((i+1) + " " +n[i]);
        }
        Console.WriteLine();

        n.Remove(2); // Deletes an item frm list
        n.Add(3) = "Daljit"; // Adds an item in the list
        n.Add(5) = "End"; // Adds in the end of the list

        Console.WriteLine ("The items in the Shopping List After modifying are : ");

        for (int i =0; i< nCount; i++)
        {
            Console.WriteLine((i+1) + " " +n[i]);
        }

        Console.ReadLine();
    }
}
```

Output:

The items in the Shopping List are : Karan Girish Neha Gaurav Raju

The items in the Shopping List After modifying are : Karan Girish Raju Daljit End

10.8 – COUNTING NUMBER OF WORDS IN A STRING

```
using System;

class CountWords
{
    public static void Main()
    {
        string s = ""; // Declare 's' of string type

        Console.Write("Enter the string : ");

        s = Console.ReadLine(); // Read string from the console & save in 's'

        string [] words = s.Split(null);
        // 'words' is an array of string type
        // string 's' will split when a space (null) is encountered
        // This will be saved into array words

        int i = words.Length; // Just count the length of array 'words'

        Console.WriteLine("The total number of words in the entered string : "+i);

        Console.ReadLine();
    }
}
```

Output:

Enter the string : Daljit is making programs

The total number of words in the entered string : 4

9.13 - REVERSING ARRAY BY CREATING A METHOD 'REVERSE'

```
using System;
public class ReverseArray
{
    public string Reverse(params string [] arr)
    {
        string [] j;
        string [] k;

        Console.WriteLine("The array list without reversing is : ");
        foreach (int i in arr)
        {
            Console.WriteLine(" "+i);
            j = new string[i]; // Save all the contents in the array 'j'
            i++;
        }

        for (int a = 0; a < j.Length ; a ++ )
        {
            k[a] = j[a]; // Saving the array in another array
        }

        for (int i = 0; i < j.Length ; i++)
        {
            j[i] = k[k.Length]; // Here we are reversing the array elements
            k.Length --;
        }

        Console.WriteLine("The reversed array now has : ");

        foreach (int i in j)
        {
            Console.WriteLine(" "+j); // Print the elements of the array 'j'
            i++;
        }
    }
}
```

10.9 - READ AN ARRAY & SORT IT

```
using System;
using System.Collections; // We need to implement collection class

class ArrayList
{
    public static void Main(string []args)
    {
        ArrayList n = new ArrayList ();

        // Read all the array items from the console
        n.Add(args[0]);
        n.Add(args[1]);
        n.Add(args[2]);
        n.Add(args[3]);
        n.Add(args[4]);

        Console.WriteLine ("The items in the Array List before sorting are : ");

        for (int i =0; i< n.Count; i++)
        {
            Console.Write (i + " : " +n[i]); // Print each array element
        }

        n.Sort(); // Sort the array list

        Console.WriteLine ("The items in the Array List after sorting are : ");

        for (int i =0; i< n.Count; i++)
        {
            Console.Write (i + " : " +n[i]); // Print each array element
        }
        Console.ReadLine();
    }
}
```

Output:

The items in the Array List before sorting are : Rajawnt Karan Girish Zeenat Daljit

The items in the Array List before sorting are : Daljit Girish Karan Rajawnt Zeenat