1D ARRAYS

Q. Create an array of length 10 of integers. Values ranging from 1 to 50.

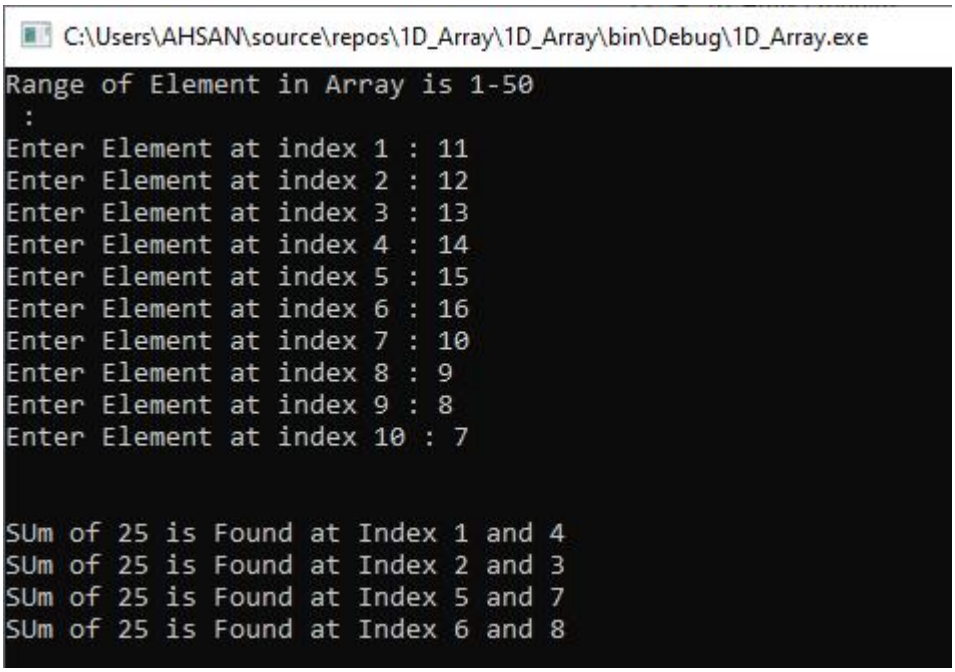
1. Find all pair of elements whose sum is 25.

SOLUTION:

```
int[] arr = new int [10];
            Console.WriteLine("Range of Element in Array is 1-50\n :");
            for (int i = 0; i < arr.Length; i++)
            {
                Console.Write("Enter Element at index {0} : ",i+1);
                arr[i] = int.Parse(Console.ReadLine());
            }
            Console.WriteLine("\n");
            int count = 0;
            for (int j = 0; j < arr.Length; j++)
            {
                for (int k = (j+1); k < arr.Length-1; k++)
                {
                    if ((arr[j]+arr[k])==25)
                    {
                        Console.WriteLine("SUm of 25 is Found at Index {0} and {1}
",j+1,k+1);
                        count++;
                    }
                }
            }
            if (count==0)
            {
                Console.WriteLine("no elements found whose sum is 25");
            }
```

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

OUTPUT:



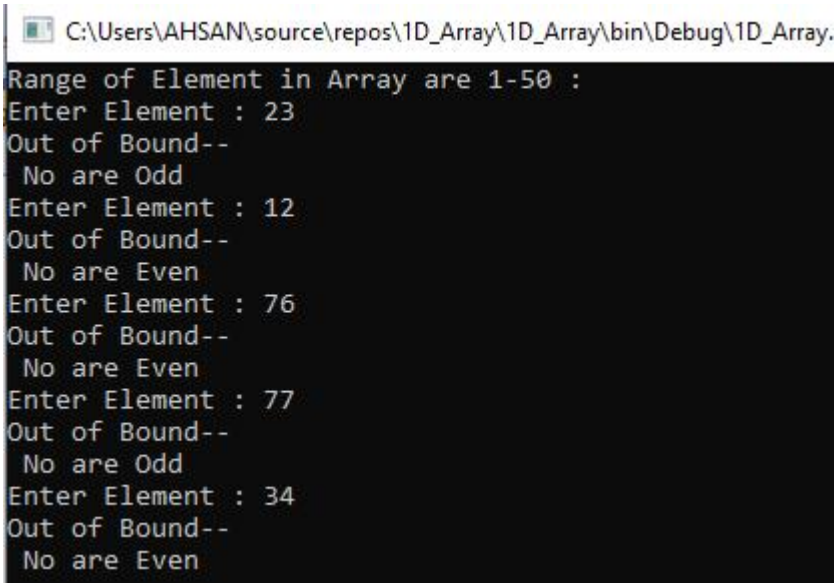
2. Find the number of elements of A which are even, and the number of elements of A which are odd.

SOLUTION:

```
int[] arr = new int [5];
            Console.WriteLine("Range of Element in Array are 1-50 :");
            for (int i = 0; i < arr.Length; i++)
            {
                Console.Write("Enter Element : ");
                int a = int.Parse(Console.ReadLine());
                if (arr[i]<1||arr[i]>50)
                {
                    Console.WriteLine("Out of Bound--");
                }
                if (a % 2 == 0)
```

```
                {
   Console.WriteLine(" No are Even");
                }
                else
                {
                    Console.WriteLine(" No are Odd");
                }
            }Console.ReadLine();
```

OUTPUT:

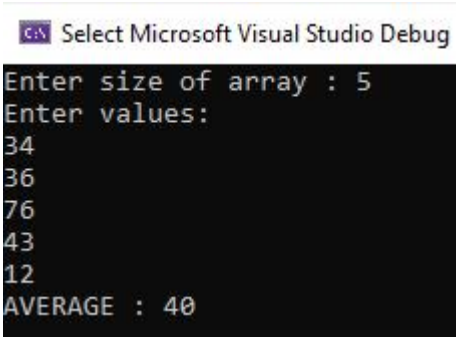


3. Write a procedure which finds the average of the value of A

SOLUTION:

```
int sum = 0;
            Console.Write("Enter size of array : ");
            int n = int.Parse(Console.ReadLine());
            int[] arr = new int[n];
            Console.WriteLine("Enter values: ");
            for (int i = 0; i < n; i++)
            {
                arr[i] = int.Parse(Console.ReadLine());
                sum += arr[i];
            }
            Console.WriteLine("AVERAGE : "+(sum/arr.Length));
```

OUTPUT:



2D ARRAYS

1. Write a program which input 2 matrix of user defined rows and columns and perform following operation

a. Display / Print as a Matrix

b. Addition of Matrix

c. Subtraction of Matrix

d. matrix multiplication

SOLUTION:

```
int a, b, i, j, m, n, x, y;
Console.Write("enter no of rows: ");
int row = int.Parse(Console.ReadLine());
Console.Write("enter no of cols: ");
int col = int.Parse(Console.ReadLine());
int[,] matrix = new int[row, col];
for (i = 0; i < row; i++)
{
    for (j = 0; j < col; j++)
    {
        Console.Write("matrix[{0},{1}]= ", i, j);
        matrix[i, j] = int.Parse(Console.ReadLine());
    }
}
Console.WriteLine();
for (i = 0; i < row; i++)
{
    for (j = 0; j < col; j++)
    {
        Console.Write(" " + matrix[i, j]);
    }
    Console.WriteLine();
}
Console.Write("\n\nenter no of rows for matrix 2: ");
int row1 = int.Parse(Console.ReadLine());
Console.Write("enter no of cols for matrix 2: ");
int col1 = int.Parse(Console.ReadLine());
```

```
            int[,] matrix1 = new int[row, col];
            for (a = 0; a < row1; a++)
            {
                for (b = 0; b < col1; b++)
                {
                    Console.Write("matrix1[{0},{1}]= ", a, b);
                    matrix1[a, b] = int.Parse(Console.ReadLine());
                }
            }
            Console.WriteLine();
            for (i = 0; i < row1; i++)
            {
                for (j = 0; j < col1; j++)
                {
                    Console.Write(" " + matrix1[i, j]);
                }
                Console.WriteLine();
            }
            m = matrix.GetLength(0);
            n = matrix.GetLength(1);
            Console.WriteLine(" \t SUM OF MATRIX ");
            int[,] sum = new int[m, n];
            for (x = 0; x < m; x++)
            {
                for (y = 0; y < n; y++)
                {
                    sum[x, y] = matrix[x, y] + matrix1[x, y];
                    Console.Write(" " + sum[x, y]);
                }
                Console.WriteLine();
            }
            Console.WriteLine(" \n\t SUBTRACT OF MATRIX ");
            int[,] sub = new int[m, n];
            for (x = 0; x < m; x++)
            {
                for (y = 0; y < n; y++)
                {
                    sub[x, y] = matrix[x, y] - matrix1[x, y];
                    Console.Write(" " + sub[x, y]);
                }
                Console.WriteLine();
            }

Console.WriteLine(" \n\t MULTIPLY OF MATRIX ");
            int[,] mul = new int[m, n];
            for (x = 0; x < m; x++)
            {
                for (y = 0; y < n; y++)
                {
                    mul[x, y] = matrix[x, y] * matrix1[x, y];
                    Console.Write(" " + mul[x, y]);
                }
                Console.WriteLine();
            }
```

OUTPUT:

e. Determinant

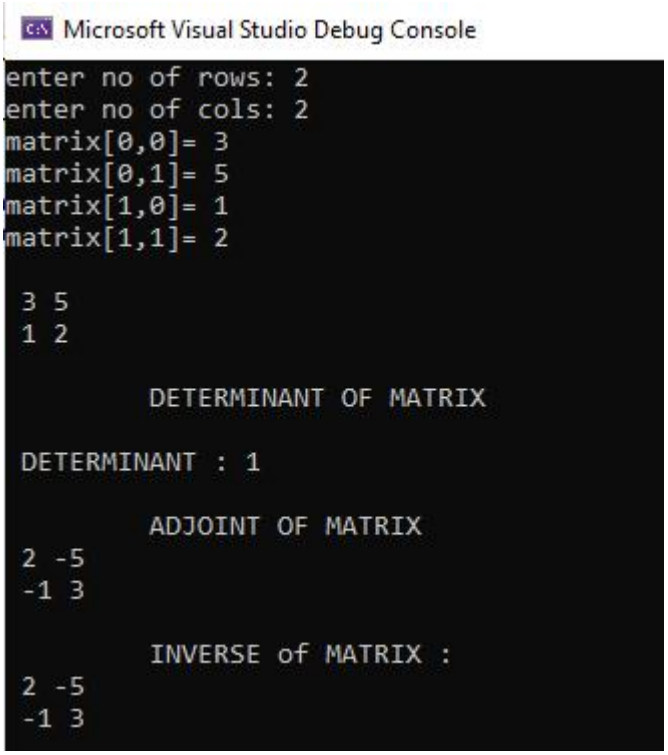
f. Inverse

SOLUTION:

```
int a, b, i, j;
            Console.Write("enter no of rows: ");
            int row = int.Parse(Console.ReadLine());
            Console.Write("enter no of cols: ");
            int col = int.Parse(Console.ReadLine());
            int[,] matrix = new int[row, col];
            for (i = 0; i < row; i++)
            {
```

```
        for (j = 0; j < col; j++)
        {
            Console.Write("matrix[{0},{1}]= ", i, j);
            matrix[i, j] = int.Parse(Console.ReadLine());
        }
    }
    Console.WriteLine();
    for (i = 0; i < row; i++)
    {
        for (j = 0; j < col; j++)
        {
            Console.Write(" " + matrix[i, j]);
        }
        Console.WriteLine();
    }
    Console.WriteLine("\n \t DETERMINANT OF MATRIX");

    a = matrix[0, 0] * matrix[1, 1];
    b = matrix[0, 1] * matrix[1, 0];
    int det = a - b;
    Console.WriteLine("\n DETERMINANT : " + det);

    Console.WriteLine("\n \t ADJOINT OF MATRIX");
    int temp;

    temp = matrix[0, 0];
    matrix[0, 0] = matrix[1, 1];
    matrix[1, 1] = temp;
    matrix[0, 1] *= -1;
    matrix[1, 0] *= -1;
    for (i = 0; i < row; i++)
    {
        for (j = 0; j < col; j++)
        {
            Console.Write(" " + matrix[i, j]);
        }
        Console.WriteLine();
    }
    Console.WriteLine("\n\t INVERSE of MATRIX : ");
    for (i = 0; i < row; i++)

    {
        for (j = 0; j < col; j++)
        {
            float d = matrix[i, j];
            float res = d / det;
            Console.Write(" " + res);
        }
        Console.WriteLine();
    }
```

OUTPUT:


Microsoft Visual Studio Debug Console
enter no of rows: 2
enter no of cols: 2
matrix[0,0]= 3
matrix[0,1]= 5
matrix[1,0]= 1
matrix[1,1]= 2
3 5
1 2
DETERMINANT OF MATRIX
DETERMINANT : 1
ADJOINT OF MATRIX
2 -5
-1 3
INVERSE of MATRIX :
2 -5
-1 3