Bahria University,

Karachi Campus

LAB EXPERIMENT NO.

12

LIST OF TASKS

TASK NO.	OBJECTIVE
1	Write a program that takes a positive integer from the console and prints the square root of this integer. If the input is negative or invalid print "Invalid Number" in the console. In all cases print "Good Bye".
2	Write a method ReadNumber(int start, int end) that reads an integer array of 10 values from the console in the range [start...end]. In case the input integer is not valid, or it is not in the required range throw appropriate exception.
3	Write a method that takes as a parameter the name of a text file then, reads the file and returns its content as string. What should the method do if an exception is thrown?

Submitted On:

January, 2022

(Date: DD/MM/YY)

Muhammad Zain | BSE - 1B

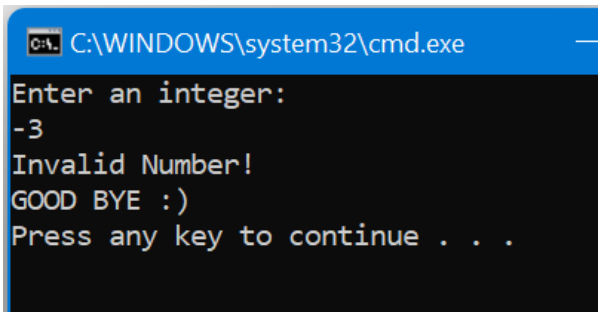
Task 1

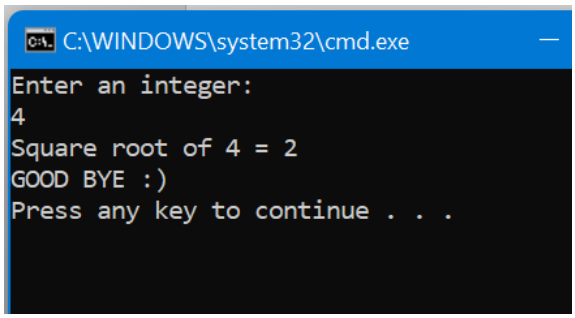
Write a program that takes a positive integer from the console and prints the square root of this integer. If the input is negative or invalid print "Invalid Number" in the console. In all cases print "Good Bye".

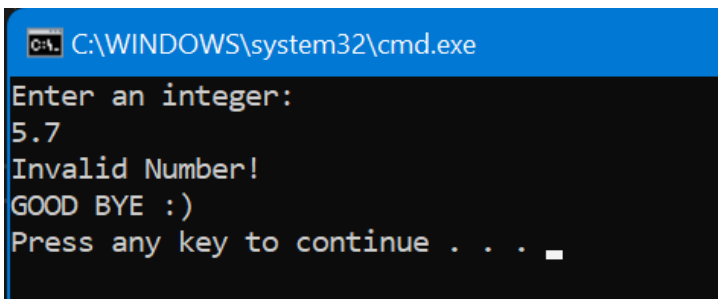
Solution:

```
using System;
namespace LAB_12
{
    class Program
    {
        static void Main(string[] args)
        {
            try
            {
                Console.WriteLine("Enter an integer: ");
                int i = int.Parse(Console.ReadLine());
                if (i >= 0)
                    Console.WriteLine("Square root of {0} = {1}", i, Math.Sqrt(i));
                else
                    Console.WriteLine("Invalid Number!");
                Console.WriteLine("GOOD BYE :)");
            }
            catch(Exception e)
            {
                Console.WriteLine("Invalid Number! \nGOOD BYE :)");
            }
        }
    }
}
```


Output:







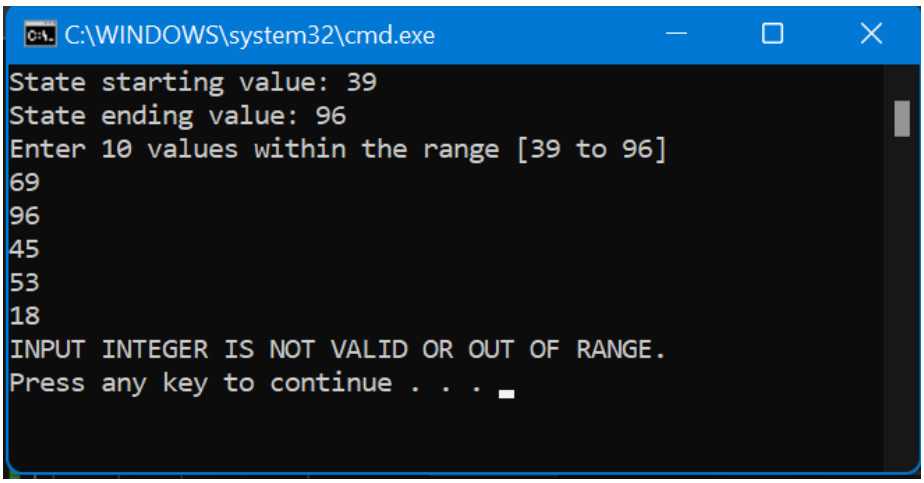
Task 2

Write a method ReadNumber(int start, int end) that reads an integer array of 10 values from the console in the range [start...end]. In case the input integer is not valid, or it is not in the required range throw appropriate exception.

Solution:

```
using System;
namespace LAB_12
{
    class Program
    {
        static void Main(string[] args)
        {
                Console.Write("State starting value: ");
                int start = int.Parse(Console.ReadLine());
                Console.Write("State ending value: ");
                int end = int.Parse(Console.ReadLine());
                ReadNumber(start, end);
        }
        public static void ReadNumber(int start, int end)
        {
            try
            {
                int[] arr = new int[10];
                Console.WriteLine("Enter 10 values within the range [{0} to {1}]", start, end);
                for (int i = 0; i < 10; i++)
                {
                    arr[i] = int.Parse(Console.ReadLine());
                    if (arr[i] >= start && arr[i] <= end)
                        continue;
                    else
                        throw new Exception("INPUT INTEGER IS NOT VALID OR OUT OF RANGE.");
                }
            }
            catch(Exception e) { Console.WriteLine(e.Message); }
        }
    }
}
```

Output:



Task 3

Write a method that takes as a parameter the name of a text file then, reads the file and returns its content as string. What should the method do if an exception is thrown?

Solution:

```
using System;
using System.IO;
using System.Text;
namespace LAB_12
{
    class Program
    {
        static void Main(string[] args)
        {
            File();
        }
        public static void File()
        {
            try
            {
                Console.Write("Enter file name: ");
                string name = Console.ReadLine();
                string filename = (name + ".txt");
                Read(filename);
            }
            catch(Exception)
            {
                Console.WriteLine("Incorrect file name");
            }
        }
        public static void Read(string filename)
        {
            var path = filename;
            var fs = new FileStream(path, FileMode.Open, FileAccess.Read);
            var sr = new StreamReader(fs, Encoding.UTF8);
            string print = sr.ReadToEnd();
            Console.WriteLine(print);
        }
    }
}
```


Output: