**PRACTICAL # 02**

**OBJECT:**

DFA implementation for integer number tokens recognition.

**THEORY:**

Compiler construction deals with the design and implementation of translator for a programming language. A compiler design is organized in phases. We will deal with the compiler front-end. This lab discusses the first phase of compiler, that is lexical analyzer.

**Program:**

This program recognizes simple integers from the code.

*using System;*

*class LexicalAnalysis {*

*public static void Main() {*

*string input = “int a=23, b; b=3+53”;*

*int i=0;*

*recognizeNum(input,* *ref i);*

*}*

*public static void recognizeNum(string src, ref int i) {*

*char peek;*

*int val = 0;*

*peek = src[i];*

*if( peek == '0' ||*

*peek == '1' ||*

*peek == '2' ||*

*peek == '3' ||*

*peek == '4' ||*

*peek == '5' ||*

*peek == '6' ||*

*peek == '7' ||*

*peek == '8' ||*

*peek == '9'*

*){*

*val = 0;*

*do{*

*val = val \* 10 + Convert.ToInt32(peek.ToString());*

*i++;*

*peek = src[i];*

*} while ( peek == '0' ||*

*peek == '1' ||*

*peek == '2' ||*

*peek == '3' ||*

*peek == '4' ||*

*peek == '5' ||*

*peek == '6' ||*

*peek == '7' ||*

*peek == '8' ||*

*peek == '9');*

*} //if digit ends here, output the number token on screen*

*if(val !=0 ){*

*Console.WriteLine("(int, {0} ) ", val);*

*val = 0;*

*i++;*

*}*

*}*

*}*

Execute the program to observe the output.

**ACTIVITIES**

**Activity 1**

Implement your own DFA for integer numbers.

**REVIEW QUESTIONS**

1. What is lexical analyzer?
2. What is a token?
3. How do you define pattern for integer tokens?