**Compiler Design (Lab 2)**

**Problem:**

One of the methods to implement a transition diagram based on DFA or NDFA has been shown in last lecture using a switch-case combination.

Another more sophisticated implementation of DFA is as follows.

s= s 0

c = nextChar();

while (c != eof){

s = move(s,c);

c = nextChar();

}

if (s is in F) return “yes”; else return “no”;

The above algorithm is applied on an input string x. The function move(s,c) gives the state to which there is an edge from state s on input c. The function nextChar() returns the next character of input string x.

Using any of the implementation provide programs to recognize following languages.

1. All strings of lowercase letters that contain the five vowels in order (a,e,i,o,u).

2. All strings of lowercase letters in which the letters are in ascending lexicographic order.

3. Comments, consisting of a string surrounded by /\* and \*/, without an intervening \*/, unless it is inside double-quotes (“).

**Input-Output:**

1. The program requires a string as an input and results in Matched if the string is recognized by the language as Not Matched.

2. The program requires a string as an input and results in Matched if the string is recognized by the language as Not Matched.

3. The program requires a string as an input and results in Matched if the string is recognized by the language as Not Matched.