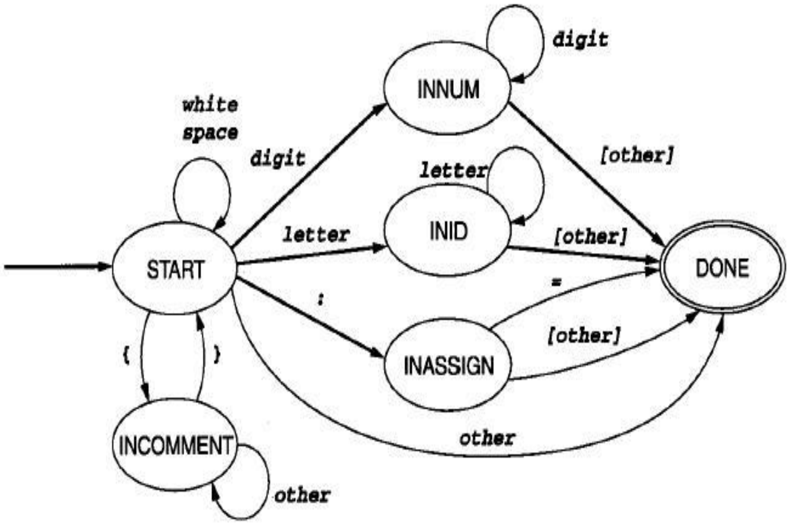
Scanner Assignment

Lexical Structure of TINY Language

Reserved Words	Special Symbols	Other
if then else end	+ - * /	number
repeat until	= <	identifier
read write	( ) ; :=	



Description

- All accepting states are collected into one state "DONE", the different token recognized is saved in a variable
- Construct a table of reserved words, reserved words are considered only after an identifier has been recognized, and then to look up the identifier in the table
- The implementation of the DFA uses the doubly nested case analysis

Implementation

- Scanner should be implemented in **C, C++, or Java** only.
- Your program should read the input from a file containing sample TINY code. The file is named **"tiny_sample_code.txt"** and you should assume that the file is in the same directory of the program and adjust the path in the code accordingly (make a relative path so that I can test without modifying your code).
- The output should be saved to a file named **"scanner_output.txt"**.

Sample Input File

```
{sample program in TINY language- computes

factorial} read x;{input an integer}

if 0<x then {don’t compute if

        x<=0} fact:=1;

        repeat

                fact:=fact*

                x; x:=x-1

        until x=0;

        write fact{output factorial of x}

end
```

Sample Output File

```
read : reserved

word x: identifier

;: special

symbol .....
```