COVER PAGE

CS323 Programming Assignments

1. Your Name [Phuc Le],
   And your partner [ ],

2. Assignment Number [One (#1)]

3. Due Date [10/2/2016]

4. Turn-In Date [10/2/2016]

5. Executable FileName [lexer.exe]

6. LabRoom [CS-200]

7. OS [Windows 10] (PC: 2a)

GRADE:

COMMENTS:

1. Problem Statement:

This project assignment is to write a lexical analyzer which input is a file containing Rat16f source code and output is a file containing series of records, and each record consists of two fields: the token and actual value of that token.

2. How to Use My Program:

Simply click on "**lexer.exe**" in **ExecutableFiles** folder, then click on 

to select the source code file; the program will display the result and write down the result to an output file.

In case the program doesn't start, there are two alternative methods to run my program:

- Method #1: copy folder **ExecutableFiles** to desktop or anywhere else. Next, get in to Windows Command line, use cd command to change to the directory to **ExecutableFiles**, and finally type "**java –jar lexer.jar**"

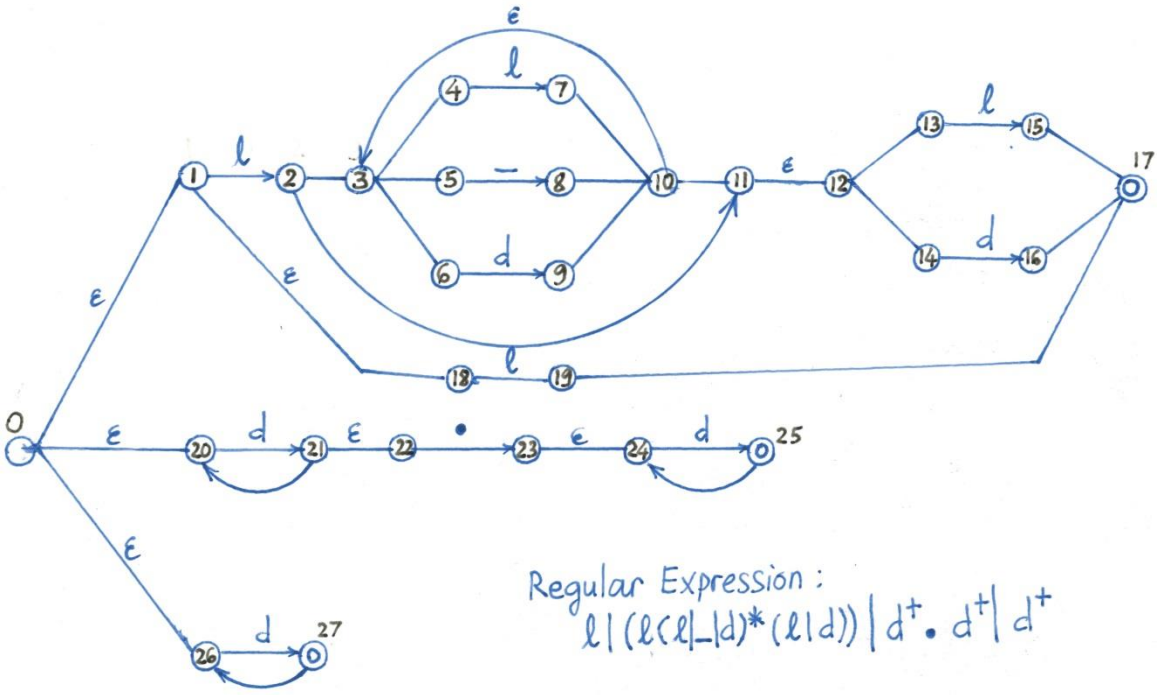
- Method #2: Open Netbeans IDE, open project "**Lexer_PhucLe**," open file "**src\cpsc323\LexicalAnalyzer.java**" and click on green button  or press **Shift-F6** to run my program

3. Designing of My Program:

I'm using least FSMs for identifiers, integer, and real, and the rest is written ad-hoc. Below is the demonstration of showing NFSM using Thompson method.

Regular Expression:

Identifiers: `l|(l(l|_|d)*(l|d))`
Reals: `d+.d+`
Integer: `d+`



Ɛ-Closure(0) = [0,1,18,20,26]	Ɛ-Closure(10) = [3,4,5,6,10,11,12,13,14]	Ɛ-Closure(20) = [20]
Ɛ-Closure(1) = [1,18]	Ɛ-Closure(11) = [11,12,13,14]	Ɛ-Closure(21) = [20,21,22]
Ɛ-Closure(2) = [2,3,4,5,6,11,12,13,14]	Ɛ-Closure(12) = [12,13,14]	Ɛ-Closure(22) = [22]
Ɛ-Closure(3) = [3,4,5,6]	Ɛ-Closure(13) = [13]	Ɛ-Closure(23) = [23,24]
Ɛ-Closure(4) = [4]	Ɛ-Closure(14) = [14]	Ɛ-Closure(24) = [24]
Ɛ-Closure(5) = [5]	Ɛ-Closure(15) = [15,17]	Ɛ-Closure(25) = [24,25]
Ɛ-Closure(6) = [6]	Ɛ-Closure(16) = [16,17]	Ɛ-Closure(26) = [26]
Ɛ-Closure(7) = [3,4,5,6,7,10,11,12,13,14]	Ɛ-Closure(17) = [17]	Ɛ-Closure(27) = [26,27]
Ɛ-Closure(8) = [3,4,5,6,8,10,11,12,13,14]	Ɛ-Closure(18) = [18]	
Ɛ-Closure(9) = [3,4,5,6,9,10,11,12,13,14]	Ɛ-Closure(19) = [17,19]	

M'	l	**d**	**_**	**.**
[0,1,18,20,26]	[2,3,4,5,6,11,13,14,17,19]	[20,21,22,26,27]	[ ]	[ ]
[20,21,22,26,**27**]	[ ]	[20,21,22,26,27]	[ ]	[23,24]
[2,3,4,5,6,11,13,14,**17**,19]	[3,4,5,6,7,10,11,12,13,14,15,17]	[3,4,5,6,9,10,11,12,13,14,16,17]	[3,4,5,6,8,10,11,12,13,14]	[ ]
[3,4,5,6,7,10,11,12,13,14,15,**17**]	[3,4,5,6,7,10,11,12,13,14,15,17]	[3,4,5,6,9,10,11,12,13,14,16,17]	[3,4,5,6,8,10,11,12,13,14]	[ ]
[3,4,5,6,9,10,11,12,13,14,16,**17**]	[3,4,5,6,7,10,11,12,13,14,15,17]	[3,4,5,6,9,10,11,12,13,14,16,17]	[3,4,5,6,8,10,11,12,13,14]	[ ]
[3,4,5,6,8,10,11,12,13,14]	[3,4,5,6,7,10,11,12,13,14,15,17]	[3,4,5,6,9,10,11,12,13,14,16,17]	[3,4,5,6,8,10,11,12,13,14]	[ ]
[23,24]	[ ]	[24,25]	[ ]	[ ]
[24,**25**]	[ ]	[24,25]	[ ]	[ ]
[ ]	[ ]	[ ]	[ ]	[ ]

Rename & Simplify:

q_o'=

N'	l	**d**	**_**	**.**
0	**2**	**1**	**8**	**8**
1	**8**	**1**	**8**	**6**
2	**3**	**4**	**5**	**8**
3	**3**	**4**	**5**	**8**
4	**3**	**4**	**5**	**8**
5	**3**	**4**	**5**	**8**
6	**8**	**7**	**8**	**8**
7	**8**	**7**	**8**	**8**
8	**8**	**8**	**8**	**8**

Finite State Machine:

M' = (∑'={ l, d, _, . },
Q'={0,1,2,3,4,5,6,7,8},
Q'_o={0}, F'={1,2,3,4,7}, N')

Source code of my program is mainly found in file **Lexer_PhucLe\src\cpsc323\MainProgram.java*

4. **Limitation:** None

5. **Shortcoming:** None