**BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE, PILANI**

**DEPARTMENT OF COMPUTER SCIENCE AND INFORMATION SYSTEMS**

**Compiler Construction (CS F363)**

**II Semester 2017-18**

**Compiler Project (Stage-1 Submission)**

**Coding Details**

**(February 26, 2018)**

1. **Personal details**

ID : 2015A7PS0040P

Name: ROHIT LODHA

1. **Files and folder details**
2. Mention the names of the Submitted files :

1\_\_driver.c\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 7\_parserDef.h\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

2\_\_\_lexer.c\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 8\_\_\_makefile\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

3\_\_lexer.h\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 9\_\_\_\_\_\_\_testcase1.txt\_\_\_\_\_\_\_\_\_\_\_\_\_

4\_\_\_parser.h\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 10\_\_\_testcase2.txt\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

5\_\_\_parser.c\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 11\_\_\_\_testcase3.txt\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

6\_\_\_\_lexerDef.h\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 12\_\_\_\_\_\_testcase4.txt\_\_\_\_\_\_\_\_\_\_\_\_\_

13\_\_\_\_testcase5.txt\_\_\_\_\_\_\_\_\_\_\_\_

14\_\_\_\_grammar.txt\_\_\_\_\_\_\_\_\_\_\_\_ 15\_\_\_\_codingDetails.docx\_\_\_\_\_\_\_\_\_\_\_\_

1. Total number of submitted files: \_\_\_\_15\_\_\_\_\_\_\_ (All files should be in ONE folder named exactly as your ID)
2. Have you compressed the folder as specified in the submission guidelines? (yes/no)\_\_\_\_Yes\_\_\_\_\_
3. **Lexer Details:**
   1. Technique used for pattern matching:   
      SWITCH CASE BASED DFA IMPLEMENTATION
   2. Keyword Handling Technique:   
      Hashing the keywords beforehand and checking for every ID detected for being a keyword
   3. Hash function description, if used for keyword handling:

djb2 by Dan Bernstein, gave unique result for keywords

* 1. Have you used twin buffer? (yes/ no) \_\_\_\_\_\_\_No\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
  2. Error handling and reporting (yes/No): Yes
  3. Describe the errors handled by you: All the Lexical errors like unidentified symbols, invalid characters in string,id,funcid, exceeding of length, invalid relational, logical operators, etc.
  4. Data Structure Description for tokenInfo (in maximum two lines):

name and type for the token; lexeme for the token and lineno for the token

1. **Parser Details:** 
   1. High Level Data Structure Description (in maximum three lines each, avoid giving C definitions used):
      1. grammar : Doubly Linked List

Type of the Grammar Node; (NT, T, |, Epsilon)

Value => In the HashTable (if NT)

name => Name of the Grammar Node (NT or T)

Next and previous pointers => to nodes of the Grammar Rule

* + 1. parse table : 2-D Array of GrammarNode  
        element pointing to the start of the rule, (size = no.of non terminals\* no of terminals)
    2. parse tree: Node: Token => for storing the token

Parent => Pointer to parent

Child => Array of pointers to child nodes

noc => No. of children

Next => Pointer to next node of the tree

ruleNode => pointer to grammar node

Tree: N-ary tree of Nodes

* + 1. Any other (specify and describe)

Rule => lineno, firstcalculated(bool) , followcalculated(bool) and pointer to head and tail of the rule, firstcalculated and followcalculated for faster calculation of subsequent first and follow set

StackNode => Same as grammar Node;

Stack => bottom and top pointer to GrammarNode

* 1. Parse tree
     1. Constructed (yes/no): yes
     2. Printing as per the given format (yes/no): yes
     3. Describe the order you have adopted for printing the parse tree nodes (in maximum two lines)

Printing inorder traversal;

First print the left child of the current node, then the currentNode and then the rest of the child, (recursion is applied)

* 1. Computation of First and Follow Sets
     1. Data structure for First and Follow sets

2-D bool matrix, rows are non terminals and columns are terminals

* + 1. FIRST and FOLLOW sets computation automated (yes /no) Yes
    2. Name the functions (if automated) for computation of First and Follow sets → findFirstSet(), findFollowSet().
    3. If computed First and Follow sets manually and represented in file/function (name that) N/A
  1. Error Handling and recovery
     1. Attempted (yes/ no): Yes
     2. Synchronizing set formation details: All the followset of the current top element of Stack and Semicolon
     3. Describe the types of errors handled : Errors like print(5) is handled; Error like “missing semicolon” before the last statement of the code is handled; Error like <string> <no> <semicolon> is handled;

1. **Compilation Details**
   1. Makefile works (yes/no):\_Yes
   2. Code Compiles (yes/ no):\_Yes
   3. Mention the .c files that do not compile:\_N/A
   4. Any specific function that does not compile:\_N/A
   5. Ensured the compatibility of your code with the specified gcc version(yes/no)\_Yes
2. **Driver Details:** Does it take care of the options specified earlier(yes/no): Yes
3. **Execution details**
   1. status (describe in maximum 2 lines): All execution works
   2. Gives segmentation fault with any of the revised test cases (1-5) uploaded on the course page. If yes, specify the testcase file name: NO
4. Specify the language features your lexer or parser is not able to handle (in maximum one line): lexer is able to handle all the feature requirements, parser may not be able to handle all the syntax errors.
5. **Lifeline detail:** Are you availing the lifeline (Yes/No): YES
6. **Declaration**: I, Rohit Lodha (your name) declare that I have put my genuine efforts in creating the compiler project code and have submitted the code developed only by me. I have not copied any piece of code from any source. If my code is found plagiarized in any form or degree, I understand that a disciplinary action as per the institute rules will be taken against me and I will accept the penalty as decided by the department of Computer Science and Information Systems, BITS, Pilani.

ID: 2015A7PS0040P

Name: Rohit Lodha

Date: 27/02/18

-------------------------------------------------------------------------------------------------------------------------------------------------

/\*not to exceed two pages\*/