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

Batch No. :

**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 \_\_\_2015A7PS0140P\_\_\_\_\_\_\_\_\_\_\_\_

Name\_\_\_\_TANVI AGGARWAL\_\_\_\_\_\_\_\_\_\_\_\_\_

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

1\_\_\_\_\_parser.c\_\_\_\_\_ 7\_\_\_\_driver.c\_\_\_\_\_\_\_\_ 13\_\_\_\_testcase4.txt\_\_\_\_\_\_\_

2\_\_\_\_\_parser.h\_\_\_\_\_\_ 8\_\_\_\_Fgrammar.txt\_\_\_ 14\_\_\_\_testcase5.txt\_\_\_\_\_\_\_

3\_\_\_\_\_parserDef.h\_\_\_ 9\_\_\_\_Ffirsts.txt\_\_\_\_\_\_\_ 15\_\_\_\_codingDetails.docx\_

4\_\_\_\_\_lexer.c\_\_\_\_\_\_\_\_ 10\_\_\_\_Ffollows.txt\_\_\_\_\_ 16\_\_\_\_makefile\_\_\_\_\_\_\_\_\_\_\_

5\_\_\_\_\_lexer.h\_\_\_\_\_\_\_\_ 11\_\_\_\_testcase2.txt\_\_\_\_\_\_

6\_\_\_\_\_lexerDef.h\_\_\_\_\_ 12\_\_\_\_testcase3.txt\_\_\_\_\_\_

1. Total number of submitted files: \_\_\_\_\_16\_\_\_\_\_ (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 and if-else
   2. Keyword Handling Technique: The distinction between keywords and ID is made by matching them with hard-coded keywords for now.
   3. Hash function description, if used for keyword handling: NA
   4. Have you used twin buffer? (yes/ no) NO
   5. Error handling and reporting (yes/No): YES
   6. Describe the errors handled by you:
      1. The length of identifiers and strings
      2. Token doesn’t match any pattern
      3. unknown symbol
   7. Data Structure Description for tokenInfo (in maximum two lines):

{ enum TokenType id; //This stores the tokenType

char \*value; //This stores the lexeme’s value

int lineNo; //This stores the corresponding line number.

}

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

**Note:** All terminals and non-terminals have been encoded using integers, to make them suitable for indexing and minimizing space used (in the case of strings).

* + 1. grammar : Grammar has been represented using an array of linked lists. The node structure has an integer(called data) and a pointer (next). Each element in the array stores corresponding Non-terminal for a rule, and a pointer to the right-hand side of the rule in a linked list (each terminal/non-terminal of the right-hand side is in a separate node).
    2. parse table: Parse Table has been represented using a 2D integer array of size (number of Non-terminals X Number of Terminals). If there is a rule corresponding to a particular pair, the parseTable stores the index of that rule in the grammar data structure.
    3. parse tree: (Describe the node structure also) Parse Tree is an n-ary tree with the following details in each node:

{ int id; //tokenType

tree \*firstChild;

tree \*siblings;

tree \*parent;

int lineNo; //line number

char \*lexeme;

}

iv. Any other (specify and describe)

Stack has been implemented to assist in parsing the input source code. Pop(), push() and push\_rhs() methods have also been implemented. The top of stack is maintained as a global pointer.

* 1. Parse tree
     1. Constructed (yes/no):\_\_\_no\_\_\_\_
     2. Printing as per the given format (yes/no): \_NA\_\_\_\_\_\_
     3. Describe the order you have adopted for printing the parse tree nodes (in maximum two lines) NA
  2. Computation of First and Follow Sets
     1. Data structure for First and Follow sets : 2D integer arrays (an entry is “1” if the terminal is present in the First/Follow sets, otherwise it is “0”).
     2. FIRST and FOLLOW sets computation automated (yes /no)\_\_\_\_\_\_\_\_\_no\_\_\_\_\_\_\_\_\_\_\_\_\_\_
     3. Name the functions (if automated) for computation of First and Follow sets\_\_\_\_\_\_\_\_NA\_\_\_\_\_\_\_\_\_\_
     4. If computed First and Follow sets manually and represented in file/function (name that) : createFirstSet() and createFollowSet() used to populate the corresponding data structures after reading from firsts.txt and follows.txt files.
  3. Error Handling and recovery
     1. Attempted (yes/ no):\_\_\_no\_\_\_\_\_\_\_\_\_
     2. Synchronizing set formation details \_\_\_\_\_\_\_\_\_\_NA\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
     3. Describe the types of errors handled :Token mismatch (expected token and input token), stack empty(input left), no rule in grammar.

1. **Compilation Details**
   1. Makefile works (yes/no):\_\_\_\_yes\_\_\_\_
   2. Code Compiles (yes/ no):\_\_\_yes\_\_\_\_
   3. Mention the .c files that do not compile: NA
   4. Any specific function that does not compile: NA
   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): removeComments and printAllTokens work for all testcases. ParseInputSourceCode works for all testcases except the first one.
   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 segmentation fault\_\_\_\_\_\_\_\_
4. Specify the language features your lexer or parser is not able to handle (in maximum one line)\_\_\_\_\_NA\_\_\_\_\_
5. **Lifeline detail:** Are you availing the lifeline (Yes/No): \_\_\_\_\_\_Yes\_\_\_\_\_\_\_\_
6. **Declaration**: I, Tanvi Aggarwal 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\_\_\_\_2015A7PS0140P\_\_\_\_\_\_

Name:\_\_TANVI AGGARWAL\_\_\_\_\_\_\_\_\_

Date: \_\_\_27-2-18\_\_\_\_\_\_

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

/\*not to exceed two pages\*/