**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 \_\_\_\_\_\_\_\_\_\_\_\_\_2014A3A70302P\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Name\_\_\_\_\_\_\_\_\_\_\_\_T DINESH RAM KUMAR\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

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

1\_\_\_\_ driver.c 7\_\_commentfree.c 13\_\_\_\_symbolhashs.h\_\_\_\_

2\_\_\_\_\_grammar-final.txt\_\_\_\_\_ 8\_\_\_\_parser.c\_\_\_\_\_\_\_ 14\_\_\_\_\_\_\_symbolhashs.c

3\_\_\_lexer.c\_\_\_\_\_ 9\_\_\_\_\_parserDef.h\_\_\_ 15\_\_codingDetails.docx\_\_\_

4\_\_\_lexer.h\_\_\_\_\_\_\_\_\_ 10\_\_\_parser.h\_\_\_\_\_\_ 16\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

5\_\_\_\_lexerDef.h\_\_\_\_\_\_ 11\_\_\_\_parserStack.h\_\_\_\_\_\_ 17\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

6\_\_\_Makefile\_\_\_\_\_ 12\_\_\_\_\_\_parserStack.c\_\_\_\_ 18\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

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: Hashing and string comparison (in some cases for finally conforming)
   2. Keyword Handling Technique: Hashing
   3. Hash function description, if used for keyword handling:Horner’s Rule based hashing
   4. Have you used twin buffer? (yes/ no) yes
   5. Error handling and reporting (yes/No):yes
   6. Describe the errors handled by you

Most of Lexical errors, Syntactic Errors

* 1. Data Structure Description for tokenInfo (in maximum two lines):

Contains token (enum), line number and union for information specific to token

1. **Parser Details:** 
   1. High Level Data Structure Description (in maximum three lines each, avoid giving C definitions used):
      1. grammar : array of rule sets ( which contain a array of rules derived by that non-terminal), first and follow sets

and some addition information.

* + 1. parse table : 2 D array of pointers to rules.
    2. parse tree: (Describe the node structure also) Child node pointer (if non-terminal), next node pointer, line number and (if terminal) lexeme related information
    3. Any other (specify and describe) stack for parsing ..
  1. Parse tree
     1. Constructed (yes/no): yes
     2. Printing as per the given format (yes/no): no (some slight modification)
     3. Describe the order you have adopted for printing the parse tree nodes (in maximum two lines)

In order traversal

* 1. Computation of First and Follow Sets
     1. Data structure for First and Follow sets : bit set operation using long int arrays (part of grammar structure)
     2. FIRST and FOLLOW sets computation automated (yes /no): yes
     3. Name the functions (if automated) for computation of First and Follow sets: computefirst, computefollow
     4. If computed First and Follow sets manually and represented in file/function (name that) \_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
  2. Error Handling and recovery
     1. Attempted (yes/ no): Yes
     2. Synchronizing set formation details : handled separately for lexer (some specific errors and others by skipping symbols) and syntax analyser (using first and follow of rules..)
     3. Describe the types of errors handled :(Parser) Missing Symbols, Unexpected Symbols

Lexer: Invalid Symbols

1. **Compilation Details**
   1. Makefile works (yes/no): yes
   2. Code Compiles (yes/ no):\_yes
   3. Mention the .c files that do not compile:\_none
   4. Any specific function that does not compile:\_\_none
   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):\_ successful
   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:\_none
4. Specify the language features your lexer or parser is not able to handle (in maximum one line)\_: It can’t recover from all kind of errors
5. **Lifeline detail:** Are you availing the lifeline (Yes/No): \_No
6. **Declaration**: I, \_\_\_\_\_\_\_T Dinesh Ram Kumar (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\_\_\_2014A3A70302P

Name:\_\_T Dinesh Ram Kumar

Date: \_26/2/2018

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

/\*not to exceed two pages\*/