BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE, PILANI
DEPARTMENT OF COMPUTER SCIENCE AND INFORMATION SYSTEMS
Compiler Construction (CS F363)
II Semester 2019-20 **Group Number: 9**
Compiler Project (Stage-1 Submission)
Coding Details
(February 24, 2020)

---

1. IDs and Names of team members

ID: 2017A7PS0057P Name: Harpinder Jot Singh

ID: 2017A7PS0080P Name: Vishal Mittal

ID:2017A7PS0072P Name: Yash Vijay

ID:2017A7PS0068P Name: Jaladi Lakshmi Teja

ID:2017A7PS0083P Name: Aditya Upadhyay

2. Mention the names of the Submitted files :

1 lexer.h	7 stackADT.h	13 hashtable.h	19 driver.c
2 lexerDef.h	8 stackADTDef.h	14 hashtableDef.h	20 tokens.txt
3 lexer.c	9 stackADT.c	15 hashtable.c	21 non_terminals.txt
4 parser.h	10 treeADT.h	16 setADT.h	22 makefile
5 parserDef.h	11 treeADTDef.h	17 setADTDef.h	23-28 testcases(t1-t6.txt)
6 parser.c	12 treeADT.c	18 setADT.c	

3. Total number of submitted files: (All files should be in **ONE folder** named exactly as Group_#, # is your group number) **28**

4. Have you mentioned your names and IDs at the top of each file (and commented well)? (Yes/ no) **Yes**

5. Have you compressed the folder as specified in the submission guidelines? (Yes/no) **Yes**

6. Lexer Details:

[A]. Technique used for pattern matching: **Switch case used to simulate the DFA**

[B]. DFA implementation (State transition using switch case, graph, transition table, any other (specify):

State transition using switch case

[C]. Keyword Handling Technique: **Hash table lookup**

[D]. Hash function description, if used for keyword handling: **summation(ASCII(i) * [pow(119,i) % 67]) % 67,**

Quadratic probing

[E]. Have you used a twin buffer? (Yes/ no) **Yes**

[F]. Lexical error handling and reporting (Yes/No): **Yes**

[G].Describe the lexical errors handled by you : 1. **symbol not in alphabet of the language, length of identifier > 20, no transition available on DFA at some input**

[H].Data Structure Description for tokenInfo (in maximum two lines): **structure with the fields token name, token value, line number. token value is an anonymous union of int, float and char***

[I]. Interface with parser : **Parser calls get_next_token() while parsing**

7. Parser Details:

[A]. High Level Data Structure Description (in maximum three lines each, avoid giving C definitions used):

i. grammar : **an array of cells, each cell contains value of lhs symbol of rule, pointer to start and end of the rule, each rule is a linked list of node containing symbol value and pointer to next node**

ii. parse table : **a two-dimensional array of integers representing the rule index**

iii. parse tree: (Describe the node structure also) : **n-ary tree with each node having fields - pointer to parent, sibling, leftmost child, rightmost child , grammar symbol, and token**

iv. Parsing Stack node structure : **dynamic array of pointers to tree nodes**

[B].Parse tree

i. Constructed (Yes/no): **Yes**

ii. Printing as per the given format (Yes/no): **Yes**

iii. Describe the order you have adopted for printing the parse tree nodes (in maximum two lines) **in-order traversal**

[C].Grammar and Computation of First and Follow Sets

i. Data structure for original grammar rules : **linked list , each node represents a grammar symbol**

ii. FIRST and FOLLOW sets computation automated (Yes /no) **Yes**

iii. Data structure for representing sets: **array of unsigned long long integers, each bit represents presence of an element**

iv. Time complexity of computing FIRST sets: **(num_of_rules)*(num_of_terminals)**

v. Name the functions (if automated) for computation of First and Follow sets : **populate_first_sets(), populate_follow_Sets()**

vi. If computed First and Follow sets manually and represented in file/function (name that):

[D]. Error Handling

i. Attempted (Yes/ no): **Yes**

ii. Printing errors (All errors/ one at a time) : **All at a time**

iii. Describe the types of errors handled :

1. **Stack's top not matching with lookahead character**
2. **Rule not found in parse table**
3. **Lexical errors**

iv. Synchronizing tokens for error recovery (describe): **follow of the non-terminal on LHS of a rule**

v. Total number of errors detected in the given testcase t6(with_syntax_errors).txt : **6 (after error recovery)**

8. Compilation Details:

[A]. Makefile works (Yes/no): **Yes**

[B]. Code Compiles (Yes/ no): **Yes**

[C]. Mention the .c files that do not compile:_

[D]. Any specific function that does not compile:_

[E]. Ensured the compatibility of your code with the specified gcc version(Yes/no): **Yes**

9. Driver Details: Does it take care of the options specified earlier(Yes/no): **Yes**

10. Execution

[A]. status (describe in maximum 2 lines): **Everything runs perfectly but parse tree is printed on console instead of a separate file**

Execution time taken for

- t1.txt (in ticks) and (in seconds) **1321 and 0.001321**
- t2.txt (in ticks) and (in seconds) **1217 and 0.001217**
- t3.txt (in ticks) and (in seconds) **1214 and 0.001214**
- t4.txt (in ticks) and (in seconds) **1616 and 0.001616**
- t5.txt (in ticks) and (in seconds) **1799 and 0.001799**
- t6.txt (in ticks) and (in seconds) **3149 and 0.003149**

[B]. Gives segmentation fault with any of the test cases (1-6) uploaded on the course page. If Yes, specify the test case file name: **No**

11. Specify the language features your lexer or parser is not able to handle (in maximum one line) **None, to the best of our knowledge**

12. Are you availing the lifeline (Yes/No): **No**

13. Declaration: We, **Harpinder Jot Singh, Jaladi Lakshmi Teja, Vishal Mittal, Aditya Upadhyay, Yash VIjay** declare that we have put our genuine efforts in creating the compiler project code and have submitted the code developed only by our group. We have not copied any piece of code from any source. If our code is found plagiarized in any form or degree, we understand that a disciplinary action as per the institute rules will be taken against us and we will accept the penalty as decided by the department of Computer Science and Information Systems, BITS, Pilani.

ID: 2017A7PS0057P Name: Harpinder Jot Singh

ID: 2017A7PS0083P Name: Vishal Mittal

ID:2017A7PS0072P Name: Yash Vijay

ID:2017A7PS0068P Name: Jaladi Lakshmi Teja

ID:2017A7PS0080P Name: Aditya Upadhyay

Date: 24 Feb 2020
