

BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE, PILANI
DEPARTMENT OF COMPUTER SCIENCE AND INFORMATION SYSTEMS

Compiler Construction (CS F363)

II Semester 2023-24

Compiler Project

Coding Details

(March 5, 2024)

Group Number

16

1. Team Members Names and IDs

ID <u>2021A7PS2690P</u>	Name <u>Sanshrav Arora</u>
ID <u>2021A7PS2542P</u>	Name <u>Ajev Malik</u>
ID <u>2021A7PS1630P</u>	Name <u>Rishabh Sahni</u>
ID <u>2021A7PS2686P</u>	Name <u>Nachiketh S Shastry</u>
ID <u>2021A7PS0533P</u>	Name <u>Rikhil Gupta</u>

2. Mention the names of the Submitted files :

1 <u>grammar.txt</u>	7 <u>parser.h</u>	13 <u>t1.txt</u>
2 <u>coding details.pdf</u>	8 <u>parser.c</u>	14 <u>t3.txt</u>
3 <u>lexerDef.h</u>	9 <u>driver.c</u>	15 <u>t5.txt</u>
4 <u>lexer.c</u>	10 <u>makefile</u>	
5 <u>lexer.h</u>	11 <u>t2.txt</u>	
6 <u>parserDef.h</u>	12 <u>t4.txt</u>	

3. Total number of submitted files (including copy the pdf file of this coding details pro forma) : 15
(All files should be in ONE folder named as Group_#)

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

5. **Lexer Details:**

[A]. Technique used for pattern matching: Pattern matching in the lexer is achieved through a finite-state machine (FSM) approach. The lexer utilizes a state transition mechanism to recognize and categorize tokens based on the input characters.

[B]. Keyword Handling Technique: Keywords are handled using a lookup table. The lexer checks whether the identified lexeme corresponds to a keyword by searching in the predefined lookup table. If a match is found, the corresponding token is assigned; otherwise, it defaults to a general identifier or field identifier token.

[C]. Hash function description, if used for keyword handling: Yes, a hash function is used for keyword handling. The hash function takes each character of the lexeme, converts it to a numerical value, and incorporates it into the hash calculation. The function ensures efficient mapping of keywords to their respective hash indices in the lookup table.

[D]. Have you used twin buffer? (yes/ no) Yes

[E]. Error handling and reporting (yes/No): Yes

[F]. Describe the errors handled by you: Token length errors: The lexer checks if the length of identifiers or field identifiers exceeds the specified limits and reports an error if necessary.

Unrecognized characters: If the lexer encounters characters that do not match any defined pattern, it reports an error.

Improper syntax or unexpected symbols: Errors are reported when unexpected symbols or syntax are encountered during tokenization.

[G].Data Structure Description for tokenInfo (in maximum two lines): The data structure tokenInfo is a structure containing information about a token, including its line number, token name (enumerated type TOKENS), and the lexeme associated with the token .

6. Parser Details:

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

- grammar : The grammar is represented as a set of production rules. Each rule consists of a non-terminal symbol followed by a sequence of terminals and/or non-terminals.
- FIRST and FOLLOW sets :These sets are implemented as linked lists of terminal symbols. Each non-terminal has a linked list of terminals in its FIRST and FOLLOW sets.
- parse table : Implemented as a 2D array where rows represent non-terminals and columns represent terminals. Each entry stores a production rule (RHS) for the corresponding non-terminal and terminal combination.
- parse tree: (Describe the node structure also) The parse tree is represented using a structure called TreeNode. Each node contains a GrammerElement (terminal or non-terminal), a pointer to the first child, and a pointer to the next sibling.
- Any other (specify and describe) : t1lexerout.txt t2lexerout.txt,t4lexerout.txt would be generated when the code runs.

[B].Parse tree

- Constructed (yes/no): Yes
- Printing as per the given format (yes/no): Yes
- Describe the order you have adopted for printing the parse tree nodes (in maximum two lines)
The parse tree nodes would be printed in Inorder

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

- Data structure for original grammar rules The grammar rules are stored in a data structure called Grammer, where each non-terminal has a linked list of RHS (Right-Hand Side) rules.
- FIRST and FOLLOW sets computation automated (yes /no) Yes
- Name the functions (if automated) for computation of First and Follow sets 'ComputeFirst' and 'ComputeFollow' functions automate the computation of FIRST and FOLLOW sets.
- If computed First and Follow sets manually and represented in file/function (name that) _____
-NA-

[D].Error Handling

- Attempted (yes/ no): Yes
- Describe the types of errors handled :Syntactical Errors

7. Compilation Details:

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

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

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

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

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

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

9. Execution

[A].status (describe in maximum 2 lines): All code is working fine.

[B].Gives segmentation fault with any of the test cases (1-6) uploaded on the course page. If yes, specify the testcase file name: t6

10. Specify the language features your lexer or parser is not able to handle (in maximum one line) Very rarely it is not able to handle token identification by the lookup table.

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

12. Declaration: We, Sanshrav Arora , Ajey Malik , Rishabh Sahni , Nachiketh S Shastry, Rikhil Gupta (your names) declare that we have put our genuine efforts in creating the compiler project code and have submitted the code developed only by us. 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 all of us in our team and we will accept the penalty as decided by the department of Computer Science and Information Systems, BITS, Pilani.

Your names and IDs

Name <u>Sanshrav Arora</u>	ID <u>2021A7PS2690P</u>
Name <u>Ajey Malik</u>	ID <u>2021A7PS2542P</u>
Name <u>Rishabh Sahni</u>	ID <u>2021A7PS1630P</u>
Name <u>Nachiketh S Shastry</u>	ID <u>2021A7PS2686P</u>
Name <u>Rikhil Gupta</u>	ID <u>2021A7PS0533P</u>

Date: 5 March 2023

Not to exceed 3 pages.