

JIET GROUP OF INSTITUTIONS

Special Examination, 2017-18

IV B. Tech. (VII Semester)

Branch: Computer Science & Engg.

7CS5: Compiler Construction

Time: 1.5 Hrs.

Max. Marks: 40

Q1. Answer the following questions:

(1X10)

1. What is a compiler?
2. What do you mean by a token?
3. What is a sentinel?
4. Write a regular expression for Identifier and keyword in Pascal.
5. What are the fundamental differences between parse trees and abstract syntax tree?
6. Describe the task performed by Loader program.
7. Define single pass and multi pass compiler.
8. Define LL (1) grammar.
9. Define the structure of Lex program.
10. A _____ is a program that matches the parameters of subroutine definition with the location of parameters.

2. Explain the different phases of compiler design with a suitable example and diagram. (10)

OR

2. Why input buffering is used in compiler? Explain Buffer pair and Sentinels input buffering techniques. (10)

3. Let G be a formal grammar with nonterminal symbols S,T,E and E^1 , terminal symbol 'x' , '+' and '\$', start symbol S, and the following production rule:

$$S \rightarrow ES$$

$$E \rightarrow TE'$$

$$E' \rightarrow +TE'$$

$$T \rightarrow X$$

- a) Construct an LL(0) parse table for the grammar calculate FIRST and FOLLOW sets as needed. (5)
- b) Use the parse table to recognize the sentence $x+x$. Show the stack and the remaining input after each step. (5)

OR

3. Construct a table-based LL(1) predictive parser for the following grammar $G = \{bexpr, \{bexpr, bterm, bfactor\}, \{not, or, and, (,), true, false\}, P\}$ with P given below:

$bexpr \rightarrow bexpr \text{ or } bterm \mid bterm$
 $bterm \rightarrow bterm \text{ and } bfactor \mid bfactor$
 $bfactor \rightarrow not \ bfactor \mid (\ bexpr \) \mid true \mid false$

For this grammar answer the following questions:

- a) Remove left recursion from G. (5)
- b) Compute the FIRST and FOLLOW sets for the non-terminals. (5)
4. Write short notes on:-
- Token, Lexemes and Pattern (5)
 - Left Recursion and Left Factoring (5)

Marking Scheme

- | | |
|---------------------------------------------------------------------|---------|
| 1. Answer the following questions:
1 for each correct option | (1X10) |
| 2. Different phases of compiler design + suitable example + diagram | 6+2+2 |
| OR | |
| 2. Input buffering + Buffer pair + Sentinels | 2+4+4 |
| OR | |
| 3. Parse table + FIRST and FOLLOW + recognize the sentence x+x | 3+2+5 |
| OR | |
| 3. Remove left recursion + FIRST and FOLLOW | 5+2+3 |
| 4. | |
| ▪ Token, Lexemes and Pattern | 2+2+1 |
| ▪ Left Recursion and Left Factoring | 2.5+2.5 |