2024

COMPUTER SCIENCE

Paper : CSMC-203

(Automata and Compiler Design)

Full Marks : 70

The figures in the margin indicate full marks.

Candidates are required to give their answers in their own words as far as practicable.

Answer ***question nos.*** **1, 2** and ***any four*** questions from the rest.

1. Answer ***any five*** questions : 2×5

(a) Verify the following identity :

$$(0 * 01 + 10) * 0 * = (0 + 01 + 10)*$$

(b) "Every unambiguous grammar is LL(1)."— Comment on the truth/falsehood of the statement.

(c) Draw the transition diagram to recognize a signed exponential number.

(d) What do you mean by pass of a compiler? How can you reduce the number of passes?

(e) Comment on : "Equivalence of PDA and CFL".

(f) Write down the conditions to be satisfied for a CFG to be in CNF.

(g) Define the dominators of a node. When is a flow graph said to be reducible?

2. Answer ***any five*** questions : 4×5

(a) Explain the meaning of handle and viable prefixes with suitable examples.

(b) Eliminate the left recursion from the following grammar :

$$S \rightarrow ABC;\ A \rightarrow Aa \mid d;\ B \rightarrow Bb \mid e;\ C \rightarrow Cc \mid f$$

(c) Generate the triple and indirect triples for the following statement :

IF A > B then C = B + D*3

else C = A + D*4

(d) For what condition a grammar can be LR(K)? How does it differ from a LL(K) grammar?

(e) What is a reserved word strategy? How is it handled in lexical analysis?

(f) What is a Right Recursive grammar? Will it create any problems in Recursive Descent Parsing? Give reasons for your answer.

(g) What input buffering concept is used in the lexical analyzer? How do the sentinels help the input buffering problem?

3. (a) Write a context-free grammar that generates all the strings of balanced parentheses.

(b) Consider the following statements :

$$G := C^*(A + B) + (A + B)$$

$$C := A + B$$

$$A := (C * D) + (E - F)$$

(i) Draw the DAG for the above statements.

(ii) What is DAG's optimal ordering for optimizing the code?

(c) Write down some of the errors that a compiler should detect. 2+6+

4. (a) Define Useful and Useless symbols in CFG. Write an algorithm to eliminate all productions containing useless symbols from the grammar.

(b) Identify Useful and Useless Symbols for the following grammar :

$$S \rightarrow aA \mid bB$$

$$A \rightarrow aA \mid a$$

$$B \rightarrow bB$$

$$D \rightarrow ab \mid Ea$$

$$E \rightarrow aC \mid d$$

(2+3)+5

5. (a) Write down the regular expression for the following :

'Set of strings consisting of an even number of a's followed by an odd number of b's.'

(b) Draw the NDFA of the above expression.

(c) Convert the above NDFA to its corresponding minimal DFA. 3+2+5

6. (a) Give the formal definition of TYPE-II grammar. Write down the TYPE-II grammar for deriving the language $\{WW^R \mid W \in (a, b)^*\}$.

(b) Simplify the following CFG and convert it to CNF :

$$S \rightarrow AaB \mid aaB$$

$$A \rightarrow \varepsilon$$

$$B \rightarrow bbA \mid \varepsilon$$

(c) What is a unit production? Why do we need to eliminate unit production from grammar? (1+3)+3+3

7. (a) Answer the following with respect to Mealy machine :

(i) "For the input string of length n, the output sequence consists of n symbols, not $n + 1$." Why is it true?

(ii) "There are no accepted states in the Mealy machine." Why is that so?

(b) Construct a Mealy machine to print out 1's complement of an input bit-stream.

(c) Consider the grammar :

$$S \rightarrow AB \mid Abad$$

$$A \rightarrow d$$

$$E \rightarrow d$$

$$E \rightarrow b$$

$$D \rightarrow b \mid \varepsilon$$

$$B \rightarrow c$$

Construct the predictive parsing table using FIRST and FOLLOW sets. Show whether the given grammar is LL(1) or not. 2+4+4

8. (a) What are the themes behind optimization techniques? Explain the flow of control optimization with an example.

(b) Construct a Syntax Directed Translation Scheme (SDTS) that translates arithmetic expressions in infix notation into arithmetic expressions in infix notation having no redundant parenthesis. Show the annotated parse tree for the input (((1 + 2) * (3 + 4)) + 5).

(c) Write down the observations of the Turing Machine in case of the following :

(i) ε — transition

(ii) Halting Problem

(iii) Infinite Loop

Give examples wherever you need them. 2+5+3

Type 0 :- $\alpha \rightarrow \beta$ $\left\{ \begin{array}{l} \alpha = V_n \\ \beta = (V_n \cup \Sigma)^* \end{array} \right.$ [V_n - Set of variable] [Σ - Set of Terminal]