M.E. ELECTRONICS AND TELE-COMMUNICATION ENGINEERING FIRST YEAR SECOND SEMESTER - 2019

Time : 3 Hours **COMPILER CONSTRUCTION (COMP)** **Full Marks :** 100

Answer any five questions ; All questions carry equal marks

1 What do we mean by dominators in connection with a flow graph? State the properties of dominators. For the flow graph shown in Fig 1 find dominators for all the nodes. 20

2. Define **IN[B] , OUT[B] , GEN[B]** and **KILL[B]** in connection with a flow graph. Consider the flow graph of Fig 2. Find IN[B] and OUT[B] for each block B1 through B5.. 20

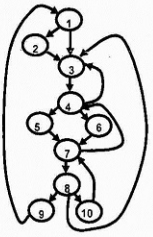

Fig 1 Flow graph

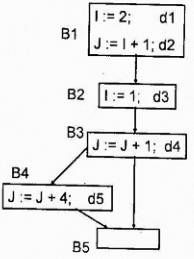

Fig 2. Flow graph

3 (a) Describe how an NFA may be derived from a given regular expression. Illustrate your answer with (a|b)* abb as an example regular expression.
(b) Derive a DFA for the NFA obtained in part(a). Also minimize the state of the DFA. 10+10

4 Define
(i) An Operator grammar
(ii) An operator precedence grammar

Show that the following grammar is an operator precedence grammar. 5+5

S → a A c B e
A → A b | b
B → d

Obtain the precedence relations for the grammar. Using these relations parse the input string "a b b c d e $" 5+5

5 (i) What are the difficulties encountered in top down parsing? Explain your answer with suitable example.
(ii) What is left factoring? Why this is necessary? Explain your answer by a suitable example in connection with top down parsing
(iii) Describe an algorithm that will eliminate left recursion from a given grammar. Use the same to eliminate the left recursion from the following grammar

S → A a | b
A → A c | S d | e

5+5+10

6 Consider the grammar
S → cAd
A → ab
A → a

For the above grammar do **any two** of the following

(i) Design the necessary procedures for the Recursive – Descent Parsing
(ii) Design SLR parser
(iii) Derive the operator precedence relations
(iv) Construct a Predictive parsing Table

Using your designed parsers parse the string "cad" 10+10

7 Consider the following grammar

B → B or T | T
T → T and F | F
F → (B) | id

where B,T and F are nonterminals and 'or' , 'and' and 'id' are terminal symbols. Eliminate the immediate left-recursion from the above grammar. Construct a predictive parsing table for the same. 10+10

8 Consider the grammar

S → AS
S → b
A → SA
A → a

(i) List all the LR(0) items for the above grammar.
(ii) Construct an NFA whose states are the LR(0) items from (i). Show that the canonical LR(0) items for the grammar is the same as the states of the equivalent DFA.
(iii) Construct the SLR parsing table for the grammar.
(iv) Parse the string "a b a b"

4×5