BACHELOR OF TECHNOLOGY (CBCS) (2020 COURSE)

B.Tech.Sem - IV Information Technology : WINTER : 2023

SUBJECT : FORMAL LANGUAGES & COMPUTATION THEORY

Day : Tuesday | Time : 02:30 PM-05:30 PM

Date : 21-11-2023 | **W-24719-2023** | Max. Marks : 60

N.B.

1) All questions are **COMPULSORY.**
2) Figures to the right indicate **FULL** marks.
3) Assume suitable data if necessary.

Q.1 a) Distinguish between Deterministic Finite Automata and Non- Deterministic Finite Automata. **(05)**

b) Draw NFA to accept binary strings containing the substring '0110'. **(05)**

OR

Q.1 a) Construct DFA for binary strings which starts with '10' and ends with '01'. **(05)**

b) Distinguish between Moore and Mealy machine with suitable example. **(05)**

Q.2 Using pumping lemma for regular sets, prove language 'L' is not regular where $L = \{a^m b^n \mid m > n\}$. **(10)**

OR

Q.2 a) Give definition of regular expression. Explain algebraic properties of R.E. **(05)**

b) Using pumping lemma prove language $L = \{0^n \mid n \text{ is a prime}\}$ is not regular. **(05)**

Q.3 Consider the grammar **(10)**

$E \rightarrow E + E \mid E * E \mid (E) \mid I$

$I \rightarrow a \mid b$

a) Show that grammar is ambiguous.

b) Remove the ambiguity and rewrite the grammar.

OR

Q.3 What is Context Free Grammar? Construct CFG corresponding to regular expression **(10)**

$R = (0+1).1^*.(1+(01)^*)$

Q.4 Construct a PDA for following CFG **(10)**

$S \rightarrow aBB$

$B \rightarrow aS \mid bS \mid a$

Test acceptance for the string '010000'.

OR

Q.4 Write short note on: **(10)**

a) Application of Context Free Grammar in compilation process

b) Removal of ambiguity in CFG

Q.5 Design a Turing Machine which reorganizes words of the form $a^n b^n c^n \mid n \geq 1$ **(10)**

OR

Q.5 Give formal definition of Turing Machine. Design a Turing Machine to find the value of $\log_2(n)$ where n is binary number. **(10)**

Q.6 Give application of Turing Machine in working of computer with suitable state diagram or flow chart. **(10)**

OR

Q.6 Explain compiler phases and application of regular expression in lexical analysis. **(10)**

* * * *