Sardar Patel Institute of Technology

Bhavan's Campus, Munshi Nagar, Andheri (West), Mumbai-400058-India
(An Autonomous Institute Affiliated to University of Mumbai)

End Semester Examination
November 2018

Max. Marks: 60 **Duration: 180 Minutes**
Class: TE **Semester: V**
Course Code:IT54/CE55 **Branch: Information Technology /Computer Engineering**
Name of the Course: Theoretical Computer Science

SYNOPTIC

Question No.		Max. Marks
Q1	a. Convert PDA to CFG and simplify the CFG and describe the langúage it accepts. (solve any one)	08
	(i). PDA is given by M= $(q_0, q_1\}, \{0,1\}, \{0,1, z_0\}, \delta, q_0, z_0, €)$, transition function δ is defined by:	
	$\delta (q_0, €, z_0) \rightarrow \{(q_1, €)\}$	
	$\delta (q_0, 0, z_0) \rightarrow \{(q_0, 0z_0)\}$	
	$\delta (q_0, 0, 0) \rightarrow \{(q_0, 00)\}$	
	$\delta (q_0, 1, 0) \rightarrow \{(q_0,10)\}$	
	$\delta (q_0,1, 1) \rightarrow \{(q_0, 11)\}$	
	$\delta (q_0, 0, 1) \rightarrow \{(q_1, €)\}$	
	$\delta (q_1, 0, 1) \rightarrow \{(q_1, €)\}$	
	$\delta (q_1, 0, 0) \rightarrow \{(q_1, €)\}$	
	$\delta (q1, €, z_0) \rightarrow \{(q_1, €)\}$	
	(ii). PDA is given by M= $(q_0, q_1\}, \{0,1\}, \{z_0, x\}, \delta, q_0, z_0, \phi)$ where δ is defined by:	
	$\delta (q_0, 1, z_0) \rightarrow \{(q_0, xz_0)\}$	
	$\delta (q_0, 1, x) \rightarrow \{(q_0, xx)\}$	
	$\delta (q_0, 0, x) \rightarrow \{(q_1, x)\}$	
	$\delta (q_0, €, z_0) \rightarrow \{(q_0, €)\}$	
	$\delta (q_1,1, x) \rightarrow \{(q_1, €)\}$	
	$\delta (q_1,0, z_0) \rightarrow \{(q_0, z_0)\}$	
	Conversion from PDA to CFG:	
	1. Production rules for start symbol	1
	2. Production rule for push operation	1
	3. Production rule for pop operation	2
	Simplification of Grammar:	
	1. Remove null production	1
		1

	2. Removing non-generating symbol 3. Removing unit production 4. Language accepted by the Grammar	1 1
	b. Construct the PDA for the following language L= {set of strings over alphabet {a, b} with exactly twice as many a's as b's} 1. Define M = (Q, Σ, q0, i/p function, stack symbol, Z_0, F) 2. Transition Diagram	04 1 3
Q2	a. Design a Turing Machine accepting following language L= The language of all non-palindromes over {a, b} 1. Define M = (Q, Σ, q0, i/p function, i/p tape symbol, B, F) 2. Transition Diagram 3. Transition function /table	08 2 3 2
	b. Write a short note on: Variants of a Turing machine (any two). Variant Turing machine 1 4. Define M for the variant chosen. 5. Working of modified Turing machine 6. Variant Turing machine 2 7. Define M for the variant chosen. 8. Working of modified Turing machine	04 1 1 1 1
Q3	a. Construct right-linear grammars for the following regular language (any one) (i). Students grades in an examination are represented with the letters {A, B, C, D, F}. A string such as ABFCAD indicated the grades obtained by a student in six different subjects. The grammar must generate only those strings that have at most three D's and no F's. (ii). Strings over {a, b} where the last two symbols in each string are a reversal of the first two symbols (i.e., last symbol = first symbol and penultimate symbol = second symbol). 1. Construction of right-linear grammar 2. Show with parsing with any sample string	08 5 3
	b. Write a short note on: Post Correspondence Problem.	04

	1. Explain PCP problem 2. Justify with example	2 2
Q4	a. What are the limitations of Finite Automata. Limitation – 1. Type of language Finite automata can accept 2. Comparison of FA with any higher-level machine.	04 1 3
	b. Construct the finite automaton DFA first and then convert it into Regular Expression for "strings over {a, b} with an odd number of a's and an odd number of b's." 1. Define DFA 2. Construction of DFA 3. Show the conversion of DFA to Regular expression.	08 1 4 3
Q5	 1. Define M=(Q, ∑, i/p function, o/p, o/p function, initial state) 2. State transition table and o/p function 3. Transition diagram 4. Moore m/c to Mealy m/c conversion with justification	06 1 2 1 2
	b. Prove that following language is not regular (i). Odd palindromes over {a, b}, that is, WCW^R, where C is a special symbol making the midpoint of the string. (ii). Binary string containing more 1s than 0s. 1. Define pumping Lemma conditions 2. Solution for L = (i) is not regular with justification 1. Define pumping Lemma conditions 2. Solution for L = (ii) is not regular with justification	04 0.5 3.5 0.5 3.5

XXX