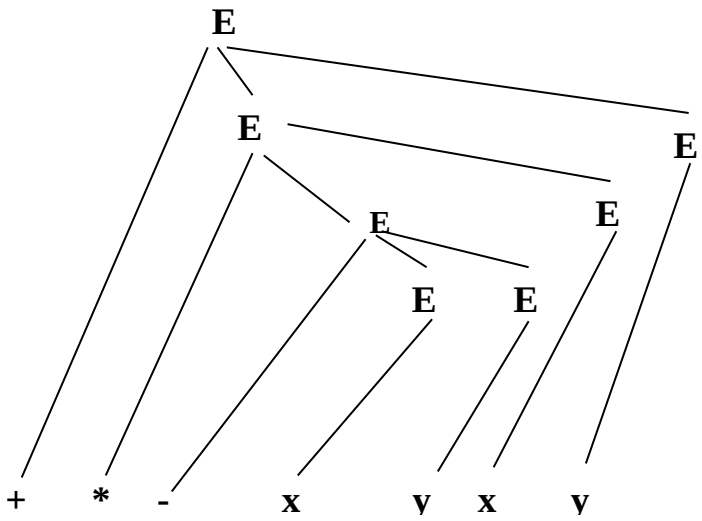
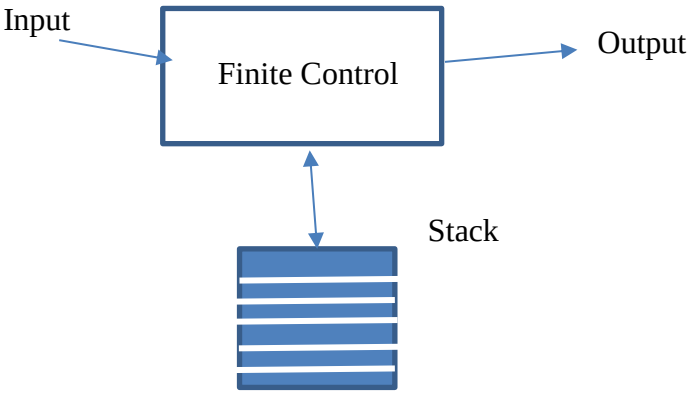
USN | | | | | | | | | | |

Internal Assessment Test 2 – Nov 2025 - SOLUTION

Sub:	Theory of Computation					Sub Code:	BCS503	Branch:	CSE
Date:	28.11.2025	Duration:	90 mins	Max Marks:	50	Sem/Sec:	5 A,B,C		OBE

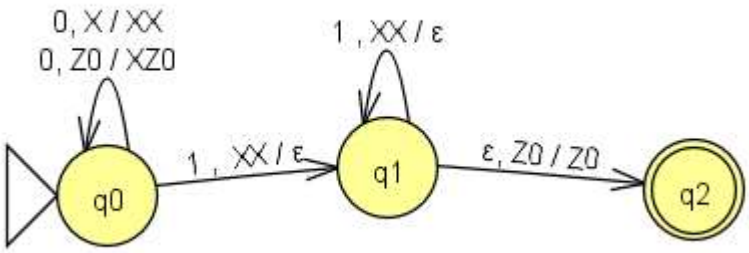
	Answer any FIVE FULL Questions	MARKS	CO	RBT
1 (a)	Consider the grammar E → + EE | *EE | -EE | x | y Find LMD, RMD, parse tree for +*-xyxy LMD E ⇒ +EE ⇒+*EEE ⇒+*-EEEE⇒+*-xEEE⇒+*-xyEE⇒+*-xyxE⇒+*-xyxy RMD E⇒+EE⇒+Ey⇒+*EEy⇒+*Exy⇒+*-EExy⇒+*-Eyxy⇒+*-xyxy Parse tree  	**7M**	**CO3**	L3
(b)	Write short notes on importance of eliminating ambiguity from grammar. When grammar is designed for a parser that parses expressions, the following need to be enforced : i) Precedence of operators should be respected, eg. * has higher precedence than +. If this precedence is not respected, the expression will be evaluated wrongly. Eg. For an expression 3+4*2 if parse tree evaluates 3+4 first then the result will be 14 as opposed to the correct answer which is 11 ii) Associativity of operators need to be respected. Some operators have a left to right associativity whilst others have a right to left associativity. Ambiguity in grammar will cause expressions to be evaluated wrongly.	**3M**	**CO3**	L3
2 (a)	Define and explain working principle of PDA.	**4M**	**CO3**	L3



A non deterministic finite automata with one additional capability where there is a stack that can store "stack symbols".
Unlike finite automaton, A PDA can remember infinite amount of information.
The finite state control reads one symbol at a time.
PDA observes symbol on top of the stack to transition based on current state, input symbol and symbol on top of stack.
In one transition, the PDA,

1. Consumes from input symbol it uses for the transition. If ε is used, no input symbol is consumed
2. Goes to new state which may or may not be the same as the previous state.
3. Replaces symbol on top of the stack by an string. An ε corresponds to popping off the top of the stack. It could be the same symbols or it can be replaced with 1 or more symbols.

(b) Design PDA for language L = $\{0^{2n}1^n \mid n \geq 1\}$. Draw transition diagram for PDA obtained. Show ID for w = 000011



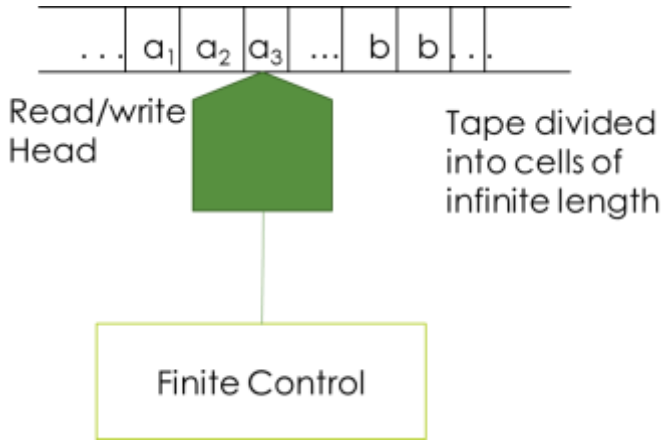
The logic is to push an X for every 0 read.
Upon seeing 1, if there are two XX on TOS, pop.
On reaching Z_0 on consuming the inputs, go to final state.

P = $(Q,\Sigma,\Gamma,\delta,q_0,Z_0,\{q2\})$
$\delta(q_0, 0,Z_0) = \{(q_0,XZ_0)\}$
$\delta(q_0, 0,X) = \{(q_0,XX)\}$
$\delta(q_0, 1,XX) = \{(q_1, \varepsilon)\}$
$\delta(q_1, 1,XX) = \{(q_1, \varepsilon)\}$
$\delta(q_1, \varepsilon,Z_0) = \{(q_2, Z_0)\}$

6M **CO3** L3

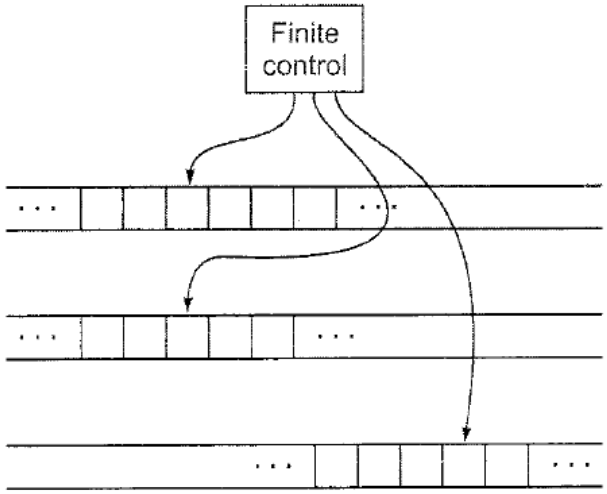
	$P = (Q,\Sigma,\Gamma,\delta,q_0,Z_0,\{q2\})$ $\delta(q_0, 0,Z_0) = \{(q_0,0Z_0)\}$, $\delta(q_0, 1,Z_0) = \{(q_0,1Z_0)\}$ $\delta(q_0, 0,0) = \{(q_0,00)\}$, $\delta(q_0,1,1) = \{(q_0,11)\}$ $\delta(q_0, 0,1) = \{(q_0,01)\}$, $\delta(q_0, 1,0) = \{(q_0,10)\}$ $\delta(q_0, c,0) = \{(q_0,0)\}$, $\delta(q_0, c,1) = \{(q_0,1)\}$, $\delta(q_0, c,Z_0)= \{(q_0,Z_0)\}$ $\delta(q_0, \varepsilon,Z_0)= \{(q_0,\varepsilon)\}$ $(q_0, 000011,Z_0) \vdash ((q_0, 00011,XZ_0) \vdash ((q_0, 0011,XXZ_0)$ $\vdash ((q_0, 011,XXXZ_0) \vdash ((q_0, 11, XXXXZ_0) \vdash ((q_1, 1,XXZ_0)$ $\vdash ((q_1, \varepsilon, Z_0) \vdash ((q_2, \varepsilon,Z_0)$			
3 (a)	Convert the following grammar to PDA. S→bABC | aBaB A→aA|bBaC|a B→bBb|a C→bCA|aAC C→d G = (V,T,S,P) = ({S,A,B,C}, {a,b,d},S,P) Let P = ({q}, T, V ∪ T, δ,q,S) that accepts via empty stack Then for every non-terminal δ(q, ε, S) = {(q,bABC), (q,aBaB)} δ(q, ε, A) = {(q,aA), (q,bBaC), (q,a)} δ(q, ε, B) = {(q,bBb), (q,a)} δ(q, ε, C) = {(q,abCA), (q,aAC), (q,d)} For every terminal δ(q, a, a) = {(q,ε)} δ(q, b, b) = {(q,ε)} δ(q, d, d) = {(q,ε)}	**5M**	**CO3**	L3
(b)	Write CFG for i) $L = \{ 0^n1^n \mid n \geq 1 \}$ S→0S1 | 01 G=(V,T,S,P) = ({S}, {0,1}, S, P) Let w=0011 S ⇒ 0S1 ⇒0011 ii) $L = \{a^nb^{n+3} \mid n \geq 1 \}$ S→aSb | abbbb Let w = aabbbbb S⇒aSb⇒aabbbbb	**5M**	**CO3**	L3
4 (a)	Define NULL production, Unit production and Useless symbol. Give one example for each. NULL production is a production of the form A→ ε where the production results in an empty string A Unit production is a production of the form A→B where a variable produces another variable. There are usually not favoured because they result in unnecessary extra steps in	**3M**	**CO3**	L1

	a derivation of a grammar. A Useless symbol is either non-generating or non-reachable. Eg. of a non-generating symbol S→AB | a A →aA B→b A is non-generating because no matter how many times A is replaced it never results in a yield. The rule S→AB also becomes useless and so does B→b Eg. of non-reachable symbol S→AB | C A→a B→b C is useless because there is no production for C.			
(b)	Define CNF. Convert the following grammar to CNF. S → ASB | Ɛ A → aAS | a B → SbS | A | bb In Chomsky normal Form(CNF) for $G = \{V, T, S, P\}$ all rules have to follow the following form: $X \longrightarrow a \quad where\ a \in \Sigma \quad OR$ $X \longrightarrow BC \quad where\ B\ and\ C\ are\ elements\ of\ V$ Every parse tree has a branching factor of 2 1. Remove ε – productions Nullable = {S} S→ASB | AB A→aAS | aA |a B→SbS | Sb | bS| b| A| bb 2. Remove Unit productions S→ASB | AB A→aAS | aA |a B→SbS | Sb | bS| b| aAS | aA |a| bb 3. Remove Mixed Productions $T_a \rightarrow a$ $T_b \rightarrow b$ S→ASB | AB A→T_aAS | T_aA |a B→ST_bS | ST_b | T_bS| b| T_aAS | T_aA |a| $T_b T_b$ 4. Remove Long Productions $T_a \rightarrow a$ $T_b \rightarrow b$ S→AM_1 | AB A→$T_a M_2$ | T_aA |a B→SM_3 | ST_b | T_bS| b| $T_a M_2$ | T_aA |a| $T_b T_b$ M_1→SB M_2→AS M_3→T_bS	**7M**	**CO3**	L3

	$G' = (V, T, S, P)$ $= (\{S, A, B, T_a, T_b, M_1, M_2, M_3\}, \{a,b\}, S, P)$			
5 (a)	Define a TM. Explain the working principle of TM with a neat block diagram. ◦ **Finite control** connected to **r/w head** ◦ Each cell can store only one symbol ◦ Input and output is effected by read/write head  A **Turing Machine** is a **7-tuple(Q,Γ, Σ, δ, q_0,b, F)** ◦ **Q**: A non-empty set of **finite states** ◦ **Γ** : a finite non-empty set of **tape** symbols ◦ **b∈ Γ** is blank ◦ **Σ** : is a non-empty set of **input symbols** and is a subset of Γ and b∉ Σ ◦ **δ** is the transition mapping (q,x) onto(q',y,D) where **D** denotes the **direction** of movement of r/w head. ◦ **D=L or D=R** according to the movement of left or right ◦ **q_0 ∈ Q** is the **initial state** ◦ **F ⊆ Q** is the set of final states. ◦ Acceptability of a string is decided by reachability from initial state to some final state.	**3M**	**CO4**	L1
(b)	Design a TM to accept the L = { WW^R| W Ɛ {a,b}*}. Show the ID for baab	**7M**	**CO4**	L3
6 (a)	State the pumping lemma for Context Free Language (CFL). Show that L= $\{a^n b^n c^n \mid n \geq 1\}$ is not context free. Let L be a CFL. Then there exists a constant k such that if w is any string in L such that if $\|w\| \geq k$, we can write w=uvxyz subject to i) $\|vxy\| \leq k$, ii) $\|vy\| \neq \varepsilon$ Since v and y has to be pumped, at least one of the strings should not be empty. iii) For all $q \geq 0$, $uv^q xy^q z$ is in L.i.e., If the strings v and y are pumped q times, including 0, the resulting string will be in L.	**6M**	**CO3**	L3

	Player 1 : The language $\{a^n b^n c^n \mid n \geq 1\}$ is not context free **Player 2** : Choose value for k $k=b^{n+1}$ Let branching factor, b=3 and number of variables, n=1 $k = 3^2 = 9$ **Player 1** : Write w such that $\|w\| \geq k$ w=aaabbbccc **Player 2** : Decide the split, such that $\|vy\| \neq \varepsilon$ and $\|vxy\| \leq k$ **Player 1** : Decide value of q, q = 2 $a^4bbbc^4 \notin L$, hence the language $\{ a^n b^n c^n \mid n \geq 1 \}$ is not context free			

a	a	a	b	b	b	c	c	c
u		v	x			y	z	

(b)	Explain Multitape Turing machine with a neat block diagram. Multi-tape ◦ Has finite set of states **Q** ◦ Initial state **q0**, ◦ Subset **F ⊆ Q** called set of final states ◦ Set of **Γ** tape symbols ◦ Symbol **b** not in **P** called blank symbol ◦ **Σ ⊆Γ**, b ∉ Σ ◦ **K** tapes, each divided into cells ◦ ***First tape*** holds input **w**. ◦ Initially, all other tape symbols hold ***blanks***. ◦ Head of tape holding input string is at the left end of the input. ◦ **Δ** is a partial function : **Q x Γ*** into **Q x Γ^k x {L,R,S}^{k}**  ◦ A move depends on current state and k tape symbols under k tape heads. ◦ In a typical move : ◦ 1) M enters a new state ◦ 2) On each tape new tape symbol is written under R/W head ◦ 3) each tape head moves {L,R,S}	**4M**	**CO4**	L1

CI **CCI** **HOD**

Course Outcomes		Blooms Level	Modules covered	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3	PSO4
CO1	Apply the fundamentals of automata theory to write DFA, NFA, Epsilon-NFA and conversion between them.	**L1, L2, L3**	**1**	3	2	2	-	2	-	-	-	-	-	-	-	2	2	-	3
CO2	Prove the properties of regular languages using regular expressions.	**L1, L2**	**2**	3	3	2	3	-	-	-	-	-	-	-	-	2	2	-	3
CO3	Design context-free grammars (CFGs) and pushdown automata (PDAs) for formal languages.	**L1, L2, L3**	**3,4**	3	3	2	3	2	-	-	-	-	-	-	-	-	2	-	3
CO4	Design Turing machines to solve the computational problems.	**L1, L2, L3**	**5**	2	3	2	3	2	-	-	-	-	-	-	-	-	2	-	3
CO5	Explain the concepts of decidability and undecidability	**L1, L2, L3**	**5**	3	2	2	3	-	-	-	-	-	-	-	-	-	2	-	3

CO PO Mapping

COGNITIVE LEVEL	REVISED BLOOMS TAXONOMY KEYWORDS
L1	List, define, tell, describe, identify, show, label, collect, examine, tabulate, quote, name, who, when, where, etc.
L2	summarize, describe, interpret, contrast, predict, associate, distinguish, estimate, differentiate, discuss, extend
L3	Apply, demonstrate, calculate, complete, illustrate, show, solve, examine, modify, relate, change, classify, experiment, discover.
L4	Analyze, separate, order, explain, connect, classify, arrange, divide, compare, select, explain, infer.
L5	Assess, decide, rank, grade, test, measure, recommend, convince, select, judge, explain, discriminate, support, conclude, compare, summarize.

PROGRAM OUTCOMES (PO), PROGRAM SPECIFIC OUTCOMES (PSO)				CORRELATION LEVELS	
PO1	Engineering knowledge	PO7	Environment and sustainability	0	No Correlation
PO2	Problem analysis	PO8	Ethics	1	Slight/Low
PO3	Design/development of solutions	PO9	Individual and team work	2	Moderate/ Medium

PO4	Conduct investigations of complex problems	PO10	Communication	3	Substantial/ High
PO5	Modern tool usage	PO11	Project management and finance		
PO6	The Engineer and society	PO12	Life-long learning		
PSO1	Develop applications using different stacks of web and programming technologies				
PSO2	Design and develop secure, parallel, distributed, networked, and digital systems				
PSO3	Apply software engineering methods to design, develop, test and manage software systems.				
PSO4	Develop intelligent applications for business and industry				