MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

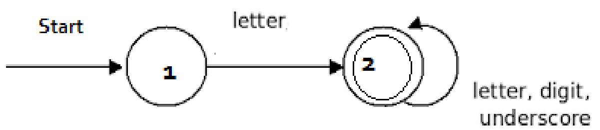
(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu.

MUST KNOW CONCEPTS	MKC

CSE	2021-22

Course Code & Course Name : 19CSC17 & THEORY OF COMPUTATION

Year/Sem/Sec : III/V/A&B

S.No.	Term	Notation (Symbol)	Concept / Definition / Meaning / Units / Equation / Expression	Units
UNIT-I : FINITE AUTOMATA				
1.	Automata Theory		Study of Abstract Machines and Automata [Self Acting Machine	
2.	Theory of computation		Branch that deals with how efficiently problems can be solved on a Abstract Machines	
3.	Finite Automaton	FA	Abstract Machines of computation used to recognize regular grammar	
4.	Formal Definition of Finite Automata		consists of the following : M= { Q, Σ, q0, F, δ } Q : Finite set of states. Σ : set of Input Symbols. q0 : Initial state. F : set of Final States. δ : Transition Function.	
5.	Types of FA		DFA- Deterministic Finite Automata and NFA/NDFA -Non-deterministic Finite Machine	
6.	Deterministic Finite Automata	DFA	For each state s and input symbol a there is at most one edge labeled a leaving s	
7.	Non-deterministic Finite Machine	NFA/NDFA	The transition from a state can be to multiple next states for each input symbol. NDFA permits empty string transitions	
8.	Minimization of DFA		Means reducing the number of states from given FA	
9.	Transition Graph		FA can be diagrammatically represented by a labeled directed graph called a transition graph	
10.	move		A state transition from one state to another on the path	
11.	FA to recognize identifier			
12.	Regular Expressions		The language accepted by finite automata can be easily described by simple expressions called Regular Expressions	
13.	Regular Expressions for Identifier		Letter(Letter/Digit)*	

14.	transition table		Tabular representation of the transition function of Automata	
15.	Types of Language		Type 0-Recursively enumerable language Type 1- Context-sensitive language Type 2- Context-free language Type 3- Regular language	
16.	Types of Automaton		Turing Machine Linear-bounded automaton Pushdown automaton Finite automaton	
17.	Equivalence of Automata in power		NFA has equal power like DFA Deterministic Pushdown automaton = NP⁄DA Deterministic Turing Machine= NTM	
18.	PDA		Pushdown automaton	
19.	TM		Turing Machine	
20.	Acceptance of Language		Recursively enumerable language accepted by TM Context-free language accepted by PDA Regular grammar accepted by FA	
21.	Hierarchy of Grammar classified by		Chomsky (1965)	
22.	Memory		FA has no memory PDA has Stack TM has Arbitrary Memory	
23.	Alphabet		is finite set of symbols. Ex : letters={a,b,c……z}	
24.	String		Finite sequence of symbols drawn from that alphabet	
25.	length of a string		Number of occurrences of symbols in s	
UNIT-II : REGULAR EXPRESSIONS AND LANGUAGES				
26.	Language		Syntactically well formed sequence of strings	
27.	Operations on Languages		Union, Concatenation, Kleen Closure, positive Closure	
28.	Algorithm used to convert RE to NFA- ε		Thompson's construction Algorithm	
29.	Algorithm used to convert NFA- ε to DFA		Subset construction algorithm	
30.	Arden's Theorem		R = Q + RP is equivalent to R = QP*	

31.	Algorithm used for Minimize DFA		Equivalence Theorem and Myhill-Nerode Theorem	
32.	Other name for Myhill-Nerode Theorem		Table Filling Method	
33.	Pumping Lemma		Used for prove that a language is not regular	
34.	Pumping Lemma worked based on		Pigeon Hole Principle	
35.	Closure Properties of Regular Languages		Regular language are closed under Union, Concatenation, Complementation, Intersection, Reversal, Difference, Homomorphism, Inverse Homomorphism	
36.	Regular Grammar		Production in the form V –> VT / T (left-regular grammar) (or)V –> TV /T (right-regular grammar)	
37.	Language for (a+b)(a+b)		{aa, ab, ba, bb}	
38.	Positive closure (a+)		One or more instances. Eg: L(a+)={a, aa,aaa, aaaa…..}	
39.	Kleen closure(a*)		Zero or more instance. Eg: L(a*)={ ε ,a, aa,aaa, aaaa…..}	
40.	L+D		Letter Union Digit Ex: {aaaa3, g8, 22aa….}	
41.	L^4		Set of all 4-letter strings.(asbc, derf, gkt…)	
42.	Language for (a+b)*		{ε, a, b, aa, ab, ba, bb, aaa, …}	
43.	a*b		String a and all strings consisting of zero or more a's and ending in b	
44.	Language for a*b		{b, ab, aab, aaab, …}	
45.	∈ - closure		The ε closure(P) is a set of states which are reachable from state P on ε-transitions.	
46.	Transition function		Movement of an automaton from one state to another for current input symbol	
47.	Transition function for DFA	δ	$\delta : Q \times \Sigma \rightarrow Q$, Q-set of states , Σ is input symbol	
48.	Transition function for NFA		$\delta: Q \times (\Sigma \cup \varepsilon) \rightarrow 2^Q$	
49.	Pushdown automaton		Finite Automata with one stack	
50.	Turing Machine		Finite Automata with two stack	
UNIT-III: CONTEXT-FREE GRAMMAR AND LANGUAGES				
51.	Context-free grammar	CFG	G = (V, T, P, S) ,V-Variable, T-Terminal P-Production, S-Start Symbol	
52.	Variable	V	Finite set of a non-terminal symbol. It is denoted by capital letters	
53.	Terminal	T	Finite set of a terminal symbol. It is denoted by lower case letters	
54.	Rule Context-free grammar		Variable→(Variable/Terminal)*	
55.	Derivations/ Parsing		The Variable in right side of the production replaced by terminal symbol called derivation	
56.	Types of Derivation		Leftmost Derivation and Rightmost Derivation	

57.	Leftmost Derivation		If the leftmost non-terminal is replaced by its production in derivation, then it called leftmost derivation	
58.	Rightmost derivation		If the rightmost non-terminal is replaced by its production in derivation, then it called rightmost derivation	
59.	Parse tree		graphical representation for the derivation of the given production rules	
60.	Properties of Parse tree		The root node is always a node indicating start symbols. The derivation is read from left to right. The leaf node is always terminal nodes. The interior nodes are always the non-terminal nodes.	
61.	Ambiguous Grammar		If there exists more than one leftmost derivation or more than one rightmost derivation called Ambiguous Grammar	
62.	Types of CFG		Chomsky Normal Form (CNF) and Greibach Normal Form (GNF)	
63.	Chomsky Normal Form	CNF	A CFG is in Chomsky Normal Form if the Productions are in the following forms • $A \rightarrow a$ • $A \rightarrow BC$ • $S \rightarrow \varepsilon$, where A, B,S and C are non-terminals and a is terminal	
64.	Greibach Normal Form	GNF	A CFG is in Greibach Normal Form if the Productions are in the following forms: $A \rightarrow b$ $A \rightarrow bD_1 \ldots D_n$ $S \rightarrow \varepsilon$ where A, $D_1,\ldots,D_n$ are non-terminals and b is a terminal.	
65.	Steps to Simplification of CFG		Elimination of Useless symbols - Unit productions- Null productions	
66.	Useless Symbols		A variable can be useless if it does not take part in the derivation of any string. That variable is known as a useless variable	
67.	Types of Useless Symbols		Non Generating symbol, Non Reachable Symbol	
68.	Non Generating symbol		If any Variable not produce terminal then it is called Non Generating symbol	
69.	Non Reachable symbol		If ay Variable not reachable from Start Symbol of the Grammar then it is called Non Reachable symbol	
70.	Unit Productions		Productions are in the following forms: $A \rightarrow B$, Where A and B is Non Terminal	
71.	ε-Production /Null production		The productions of type $S \rightarrow \varepsilon$ are called ε productions	
72.	Remove Unit Production in $S \rightarrow 0A \mid 1B \mid C$ $C \rightarrow 01$		$S \rightarrow 0A \mid 1B \mid 01$ $C \rightarrow 01$	
73.	Remove Null Production in $S \rightarrow S0A \mid 1BS \mid \varepsilon$		$S \rightarrow S0A \mid 1BS \mid 0A \mid 1B$	
74.	Left Linear Grammar		Productions are in the following forms: $A \rightarrow Ba$, Where A and B is Non Terminal	

75.	Right Linear Grammar		Productions are in the following forms: A → aB, Where A and B is Non Terminal	
		Unit-IV :PUSHDOWN AUTOMATA		
76.	Pushdown automata	PDA	recognize CFG (Context free Grammar)	
77.	In Power		PDA is more powerful than FA TM is more powerful than PDA	
78.	Stack in PDA		Used to provide a last-in-first-out memory management capability to Pushdown automata	
79.	Function in PDA		PDA can push an element onto the top of the stack and pop off an element from the top of the stack.	
80.	Formal definition of PDA		M={ Q, ∑, Γ, δ, q0, Z, F} Q: Finite set of states ∑: Input set Γ: Stack symbol q0: Initial state Z: Start symbol of the stack. Γ. F: Final states δ: Transition /Mapping function	
81.	Instantaneous Description of PDA		The Execution status of the PDA at any time can represented by the instantaneous description (ID) of a PDA, It is represented by a triplet (q, w, s) where • q is the current state • w is the string to be processed by the PDA • s is the stack contents	
82.	Moves of PDA		Moves of PDA from current configuration to next configuration can be represented by the symbol $\vdash$ The "turnstile" notation. (p, aw, Tβ) $\vdash$ (q, w, αb)	
83.	Operation of the stack in PDA		Push Pop Skip	
84.	Types of PDA		Deterministic pushdown automata and Non- Deterministic pushdown automata	
85.	Language accepted by Pushdown automata		Acceptance by Final State Acceptance by Empty Stack	
86.	CFL		Context free Language	
87.	Pumping lemma for CFL		To prove that a language L is not context free called Pumping lemma	
88.	Closure Properties of CFL		CFL are closed under Union, Concatenation, Closure, Reversal, Difference, Homomorphism, Inverse Homomorphism	
89.	CFL are not closed under		Complementation, Intersection, Difference,Subset	
90.	CFL Language eg		L={a^nb^n/n>=0}, L={$a^{2n}b^n$/n>=0}, L={a^nb^{2n}/n>=0}, L={WW^r/ w=[a-z]}	
91.	Non-CFL Language eg		L={a^n/n>=0}, L={a2^n/n>=0}, L={a^n/n is prime}, L={WW/w=[a-z] }	
92.	Moore Machine		Moore machine is a finite-state machine whose output values are determined only by its current state	

93.	Mealy Machine		Mealy machine is a finite-state machine whose output values are determined both by its current state and the current inputs	
94.	Stack		A push down automaton employs _________ data structure.	
95.	Type 2-CFL		Push down automata accepts __________ languages.	
96.	Counter Automaton		A push down automaton with only symbol allowed on the stack along with fixed symbol	
97.	Relation to Chomsky hierarchy		Regular<CFL<CSL<Unrestricted	
98.	strings generated by the given grammar: S->SaSbS|e		aabb abab abaabb	
99.	X->aX		Right Recursive Grammar	
100.	pumping lemma for the context free languages		uv^nwx^ny	
Unit-V:TURING MACHINES & UNDECIDABILITY				
101.	Turing machines		Turing machines Accepts Recursive enumerable language	
102.	Language accepted by a turing machine is called _____________		Recursive Ennumerable and Recursive	
103.	Founder of turing machine		Alan Turing in 1936	
104.	Definition of Turing machines		7-tuple $(Q, X, \sum, \delta, q_0, B, F)$ • Q is a finite set of states • X is the tape alphabet • ∑ is the input alphabet • δ is a transition function; $\delta : Q \times X \rightarrow Q \times X \times$ {Left_shift, Right_shift}. • q_0 is the initial state • B is the blank symbol • F is the set of final	
105.	Halting Problem		Halting means that the program on certain input will accept it and halt or reject	
106.	Recursive Language		Language Halts either in accept or reject	
107.	Recursive enumerable language		Language accept or enter to loop	
108.	Decidable Problems		A problem is decidable if we can construct a Turing machine which will halt in finite amount of time	
109.	Undecidable Problems		If there is no Turing machine which will always halt in finite amount of time to give answer as 'yes' or 'no'	
110.	Rice's Theorem		Every non-trivial (answer is not known) problem on Recursive Enumerable languages is undecidable	
111.	Post's Correspondence Problem		Arrange tiles in such order that string made by Numerators is same as string made by Denominators	
112.	P-Problem		P is set of problems that can be solved by a deterministic Turing machine in **P**olynomial time.	
113.	NP-Problem		NP is set of decision problems that can be solved by a **N**on-deterministic Turing Machine in **P**olynomial time	
114.	NP- Completeness		If $L \in NP$ and L is NP-hard L-is the Language	

115.	NP-Hard		NP-hard if for all Language ϵ NP, we can solve L in polynomial time, we can solve all NP problems in polynomial time	
116.	A turing machine operates over		Infinite memory tape	
117.	Turing Completeness		The ability for a system of instructions to simulate a Turing Machine is called _________	
118.	Universal Turing machine		A turing machine that is able to simulate other turing machines	
119.	Multi-tape turing machine		A turing machine with several tapes	
120.	Diagonalization		Technique is used to find whether a natural language isn't recursive enumerable	
121.	Rice's theorem		Rice's theorem states that 'Any non trivial property about the language recognized by a turing machine is undecidable	
122.	Trivial		A property of partial functions is called trivial if it holds for all partial computable functions or for none	
123.	NP stands for		Non-deterministic polynomial	
124.	NP		Travelling sales man problem belongs to which of the class	
125.	$O(n^k)$, k∈N		NP problems are the set of decision problems which can be solved using a non deterministic machine in _______ time	
		Placement Questions		
126.	Which of the following cannot be solved using polynomial time		Linear Programming Greatest common divisor Maximum matching	
127.	P-complete type of problem		Circuit Value problem Linear programming Context free grammar membership	
128.	A problem which is both _______ and _________ is said to be NP complete		NP, NP hard	
129.	Post Correspondence problem is	PCP	Undecidable decision problem	
130.	tractable		A problem is called __________ if its has an efficient algorithm for itself.	
131.	Runtimes of efficient algorithms		$O(n)$, $O(nlogn)$, $O(n^3log2^n)$	
132.	Runtimes of inefficient algorithms		$O(2^n)$, $O(n!)$	
133.	polynomial		An algorithm is called efficient if it runs in ____________ time on a serial computer.	
134.	Halting problem		Is undecidable	
135.	Example of undecidable problems		Determining whether a grammar is ambiguous and two grammars generate the same language	
136.	Which of the games fill under the category of Turing-complete		Minecraft Minesweeper Dwarf Fortress	
137.	an enumerator enumerates it		A language is turing recognizable if an only if ___________	

DESIGNING YOUR FUTURE
Estd. 2000

138.	an output printer		Enumerator is a turing machine with ___________	
139.	accepted		Every language accepted by a k-tape TM is _____ by a single-tape TM	
140.	NO		Can a multitape turing machine have an infinte number of tapes	
141.	n-track Turing machine		In a n-track Turing machine, one head reads and writes on all tracks simultaneously	
142.	universal state		accepting if every transition leads to an accepting state	
143.	existential state		accepting if some transitions leads to an accepting state	
144.	Alternating Turing machine		Which of the turing machines have existential and universal states	
145.	Read-only turing machine		Which of the following is not a Non deterministic turing machine	
146.	Equal		A multitape turing machine is ________ powerful than a single tape turing machine	
147.	functions can a turing machine perform		Copying a string Deleting a symbol Accepting	
148.	functions can a turing machine not perform		Inserting a symbol	
149.	Which of the following can accept even palindrome over		Turing machine	
150.	A language L is said to be Turing decidable if		Recursive & TM recognizes L	

Faculty Team Prepared **Signatures**

1. **M.Ganthimathi**
2. **M.Azhagesan** **HoD**

DESIGNING YOUR FUTURE
Estd. 2000