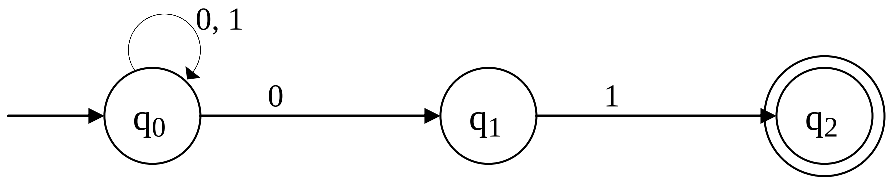
Sreenivasa Institute of Technology and Management Studies (Autonomous)
Chittoor - 517127

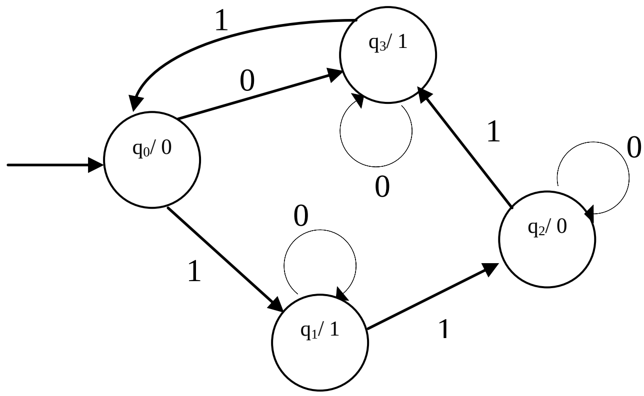
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Year / Sem : II / II
Sub. Code & Subject : 18CSE225 – Formal Languages and Automata Theory
Prepared by : Dr. D. Jagadeesan, Professor / CSE

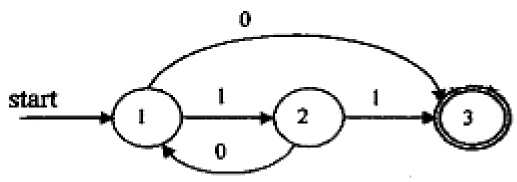
QUESTION BANK

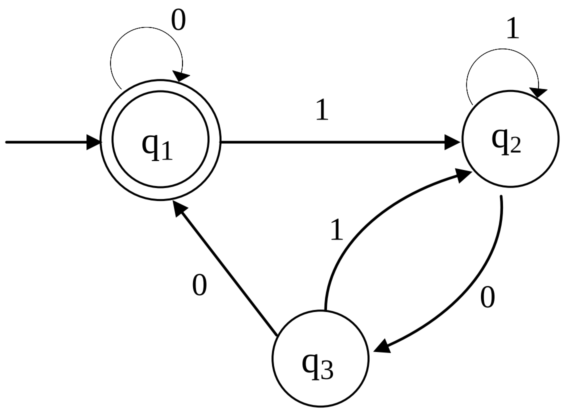
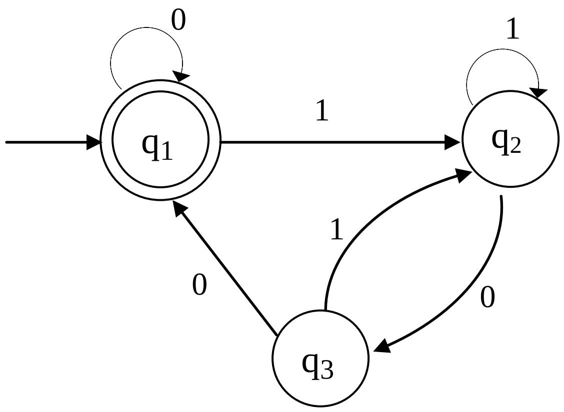
Unit – I

Sl. No	Questions	CO	PO	BT
	Part - A			
1	Write down the operations on set.	1	1	1
2	List any three applications of Automata Theory.	1	1	1
3	Define Finite Automation.	1	1	1
4	Define Deterministic Finite Automation.	1	1	1
5	Define Non-Deterministic Finite Automation.	1	1	1
6	Define NFA with ε transition.	1	1	1
7	Design FA which accepts odd number of 1's and any number of 0's.	1	2,3	6
8	Design FA to check whether given unary number is divisible by three.	1	2,3	6
9	Design FA to check whether given binary number is divisible by three.	1	2,3	6
10	Design FA to accept the string that always ends with 00.	1	2,3	6
11	Obtain the ε closure of states q0 and q1 in the following NFA with ε transition.	1	2	5
12	Obtain ε closure of each state in the following NFA with ε move.	1	2	5
13	Explain a transition diagram.	1	1	2
14	Explain a transition table.	1	1	2
15	Explain the transition function.	1	1	2
16	Differentiate DFA and NFA?	1	2	2
17	Write notes on Moore Machine.	1	1	6
18	Write the formal definition of Moore Machine.	1	1	6
19	Short notes on Mealy Machine.	1	1	1
20	Write the formal definition of Mealy Machine.	1	1	6
21	Compare the Mealy and Moore Model?	1	2	5
	Part – B			
22	Design FA to accept the string that always ends with 00.	1	2,3	6
23	Design FA to check whether given binary number is divisible by three.	1	2,3	6
24	Show that "For every NFA, there exists a DFA which simulates the behavior of NFA. If L is the set accepted by NFA, then there exists a DFA which also accepts L".	1	2	1
25	Show that "If L is accepted by NFA with ε-moves, then there exists L which is accepted by NFA without ε-moves.	1	2	1
26	Construct DFA equivalent to the given NFA 	1	2,3	6

<table>
<tr><td>27</td><td>Let M = ({q0, q1}, {0,1}, δ, q0, {q1}) be NFA. Where δ (q0, 0) = {q0, q1}, δ (q0, 1) = {q1}, δ (q1, 0) = {ϕ}, δ (q1, 1) = {q0, q1}. Construct its equivalent DFA.</td><td>1</td><td>2,3</td><td>6</td></tr>
<tr><td>28</td><td>Let M = ({q0, q1,q2,q3}, {0,1}, δ, q0, {q2,q3}) be ε-NFA.
Where δ (q0, 0) = {q0, q1}, δ (q0, 1) = {q1}, δ (q1, 0) = {q2,q3}, δ (q1, ε) = {q1}, δ (q1, 1) = {q0, q1}, δ (q2, 0) = {q2}, δ (q2, ε) = {q3}, δ (q2, 1) = {q0, q3,}, δ (q3, 0) = {q3}, δ (q3, 1) = {q2, q3}, δ (q3, ε) = {q0}. Construct its equivalent DFA.</td><td>1</td><td>2,3</td><td>6</td></tr>
<tr><td>29</td><td>Consider the following ε–NFA. Compute the ε–closure of each state and find it's equivalent DFA.
<table>
<tr><td></td><td>ε</td><td>a</td><td>b</td><td>c</td></tr>
<tr><td>p</td><td>Φ</td><td>{p}</td><td>{q}</td><td>Φ</td></tr>
<tr><td>q</td><td>{p}</td><td>{q}</td><td>{r}</td><td>Φ</td></tr>
<tr><td>*r</td><td>{q}</td><td>{r}</td><td>ϕ</td><td>{p}</td></tr>
</table></td><td>1</td><td>2,3</td><td>5</td></tr>
<tr><td>30</td><td>Convert a NFA which accepts the string ends with 01 to a DFA.</td><td>1</td><td>2,3</td><td>5</td></tr>
<tr><td>31</td><td>Consider the Moore machine described by the transition diagram given below. To construct a Mealy machine, which is equivalent to moore machine

</td><td>1</td><td>2,3</td><td>5</td></tr>
<tr><td>32</td><td>Consider the Mealy machine described by the transition table given below. To construct a Moore machine, which is equivalent to mealy machine?
<table>
<tr><th rowspan="2">Present State</th><th colspan="2">input = 0</th><th colspan="2">input = 1</th></tr>
<tr><th>Next State</th><th>Output</th><th>Next State</th><th>Output</th></tr>
<tr><td>→q1</td><td>q3</td><td>0</td><td>q2</td><td>0</td></tr>
<tr><td>q2</td><td>q1</td><td>1</td><td>q4</td><td>0</td></tr>
<tr><td>q3</td><td>q2</td><td>1</td><td>q1</td><td>1</td></tr>
<tr><td>q4</td><td>q4</td><td>1</td><td>q3</td><td>0</td></tr>
</table></td><td>1</td><td>2,3</td><td>5</td></tr>
</table>

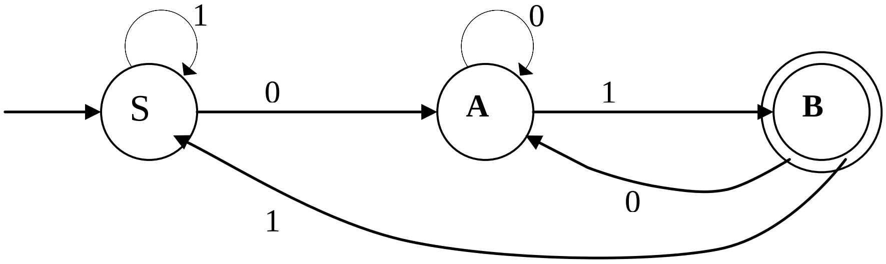
Unit – II

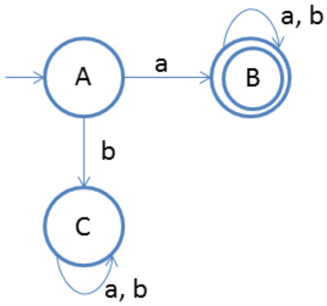
Sl. No	Questions	CO	PO	BT
	Part - A			
1	State regular expression.	2	1	1
2	How the kleen's closure of L can be denoted?	2	1,2	4
3	How do you represent positive closure of L?	2	1,2	4
4	Write the regular expression for the language accepting all combinations of a's over the set $\sum = \{a\}$.	2	2,3	6
5	Write regular expression for the language accepting the strings which are starting with 1 and ending with 0, over the set $\sum = \{0,1\}$.	2	2,3	6
6	Show that (0*1*)* = (0+1)*.	2	2	2
7	Show that (r+s)* $\neq$ r* + s*.	2	2	2
8	If L = { The language starting and ending with 'a' and having any combinations of b's in between, that what is r?	2	2,3	4
9	Give regular expression for L= L1 $\cap$ L2 over alphabet {a,b} where L1 = all strings of even length L2 = all strings starting with 'b'.	2	2,3	2
10	Explain the application of the pumping lemma.	2		3
11	Describe the following by regular expression a. L1 = the set of all strings of 0's and 1's ending in 00. b. L2 = the set of all strings of 0's and 1's beginning with 0 and ending with 1.	2	2,3	1
12	Show that (r*)* = r* for a regular expression r.	2	2	2
13	Write down the closure properties of regular language.	2	3	6
14	What is pumping lemma?	2	2	4
15	State Arden's theorem.	2	1	1
16	What is dead state?	2	2	4
	Part – B			
17	Show that 'r' be a regular expression, the there exists an NFA with ε transitions that accepts L{r}.	2	2	2
18	Construct the NFA with ε for the regular expression using Thomson construction method. a. 0(0+1)*100 b. a(a+b)*b	2	2,3	6
19	Obtain the equivalent DFA from the following regular expressions a. (a+b)*abb b. (00+11)*(0+1)*	2	2,3	5
20	Show that the following languages are not regular using pumping lemma a. L={$0^i 1^i$; i>=1} b. L={a^p ; p is prime}	2	2,3	2
21	Find the regular expression for the set of all strings denotes by $(R_{13})^2$ from the deterministic finite automata given below  	2	2,3	5

<table>
<tr><td>22</td><td>Obtain the regular expression using Arden’s theorem from the given DFA.

</td><td>2</td><td>2,3</td><td>5</td></tr>
<tr><td>23</td><td>Obtain the regular expression using state elimination method from the given DFA.

</td><td>2</td><td>2,3</td><td>5</td></tr>
</table>

Unit – III

Sl. No	Questions	CO	PO	BT
	Part - A			
1	Obtain the Right Linear Grammar from the given Left Linear Grammar	3	1	5
2	Let G = ({S,C}, {a,b}, P,S} where P consists of S → aCa, C→ aCa, Find L(G))?	3	2	5
3	Consider G whose productions are S→ aAS/ a ,A→SbA / SS / ba, show that S→ aabbaa and construct a derivation tree.	3	2	2
4	Find L(G) where G = ({S},{0,1}, {S→0S1, s→ε },S)	3	2	5
5	Construct a CFL from the given grammar S→ aaA, A→S / a	3	2	6
6	Define a derivation tree for CFG.	3	1	1
7	Construct CFG L= { a^nb^n ; n≥1}.	3	2,3	6
8	Find a LM derivation for aaabbabbba with the productions.	3	2	5
9	Find L(G), S → aSb, S → ab.	3	2	5
10	Show that id* id can be generated by two distinct leftmost derivation in the grammar	3	2	2

11	Write a CFG for the set of strings which does not produce any palindromes.	3	2	6
12	Find the derivation tree for the grammar G = ({S, A, B}, {a,b}, P, S} Where P is given by S → Aa / bB A→ab B→ aBb / a	3	2	5
13	Define parse tree.	3	1	1
14	What are the two major normal forms for context-free grammar?	3	2	4
15	What is a useless symbol?	3	2	4
16	Define Nullable Variable?	3	1	1
17	Let G = (V, T, P,S) with the productions given by S → aSbS/ B / ε B→ abB Eliminate the useless production.	3	2	5
18	What is a useful production?	3	2	4
19	Determine whether the grammar G has a useless production? S→A A→aA / ε B → bA	3	2	4
20	Write a procedure to eliminate ε production.	3	2	6
21	Write the procedure to eliminate the unit productions.	3	2	6
22	Define CNF.	3	1	1
23	Define GNF.	3	1	1
	Part – B			
24	Consider the Grammar G whose productions are S → 0B / 1A A → 0 / 0S / 1AA B → 1 / 1S / 0BB and the string 0110 a. Find the left most derivation and associated derivation tree. b. Find the right most derivation and associated derivation tree. c. Show that the G is ambiguous. d. Find L(G)	3	2	5
25	Consider the Grammar whose productions are S → aAS / a A → SbA / SS / ba a. Construct a LMD and RMD Tree for S =>* aabbaa b. Find the above grammar is ambiguous or unambiguous.	3	2	5
26	Construct Right Linear Grammar from the given Finite Automata 	3	2,3	6

27	Construct Left Linear Grammar from the given Finite Automata  	3	2,3	6
28	Construct a Finite Automata from the following Right Linear Grammar a) A → aB/bA/b B → aC/bB C → aA/bC/a b) S → A / B / ε A → /1B/0 B → 0S/1A/1	3	2,3	6
29	Construct a Finite Automata from the following Left Linear Grammar a) A → Ba/Ab/b B → Ca/Bb C → Aa/Cb/a b) S → Aab / Aba / B / ε A→ Sb / b B → Sa C → ε	3	2,3	6
30	Consider the grammar S→ 0A0 / 1B1 / BB A→ C B→ S / A C→ S / ε and simplify using the same order a. Eliminate ε-Productions b. Eliminate unit productions c. Eliminate useless symbols	3	2,3	5
31	Let G be the following grammar with productions: a) S → bA / aB A → bAA / aS / ε B → aBB / bS / b. Convert the above grammar G into CNF b) S→ 0A0 / 1B1 / BB A→ C B→ S / A C→ S / ε. Convert the above grammar G into CNF	3	2,3	5
32	Find the GNF equivalent of the given grammar S → AA / 0 A→ SS / 1	3	2,3	5
33	Consider the Grammar G = ({S,A,B},{a,b}, P, S} as the productions S → AB A → BS / b B → SA / a Convert it into GNF.	3	2,3	5

Unit – IV

Sl. No	Questions	CO	PO	BT
	Part – A			
1	Define pushdown automaton.	4	1	1
2	What are the different ways of language acceptances by a PDA and define them.	4	2	4
3	Construct a PDA that accepts the language generated by the grammar S→ aSbb / aab	4	2,3	6
4	Construct a PDA that accepts the language generated by the grammar S → aABB , A → aB / a , B → bA / b	4	2,3	6
5	How do you convert CFG to a PDA.	4	2	6
6	Define Deterministic PDA.	4	1	1
7	Is it true that NDPA is more powerful than that od DPDA? Justify your answer.	4	2	5
8	Is it true that the language accepted by a PDA by empty stack and final states are different languages.	4	2	5
9	What is the additional feature PDA has when compared with NFA? Is PDA superior over NFA in the sense L acceptance? Justify your answer.	4	2	4
	Part – B			
10	Prove that if L=N(PN) for some PDA PN = (Q, Σ, Γ, δ, q0, Z0, F), then there is a PDA PF such that L=L(PF).	4	1,2	2
11	Prove that if M1 = (Q, Σ, Γ, δ, q0, Z0, F) accept by final state, we can find a PDA M2, accepting L by empty store i.e., L = L(M1) = N(M2).	4	1,2	2
12	Construct a PDA that accepts the following languages a. L= {wcwr | W in (0+1)* } by empty stack or final state b. L = {wwR ; w ∈ (0+1)*} by empty stack or final state c. L = {0n 1n ; n ≥ 0} accepted by empty stack or final state d. L = {anbmcmdn; n, m ≥ 1} accepted by empty store and check whether the string w = aaabcddd is accept or not.	4	2,3	6
13	Construct a PDA that will accept the language generated by the grammar G = ({S, A}, {a, b}, P, S) with the productions S → AA / a, A → SA / b and test whether "abbabb" is in N(M).	4	2,3	6
14	Consider the grammar G = (V, T, P, S) and test whether "0101001" is in N(M). Where Productions are S → 0S/1A/1/0B/0 A → 0A/1B/0/1 B → 0B/1A/0/1	4	2,3	4
15	Construct a PDA from the given CFG G = ({S, A}, {a, b}, P, S) where the productions are S → AS / ε and A → aAb / Sb / a	4	2,3	6
16	Construct a PDA from the following CFG. G = (V, T, P, S) with V = {S}, T = {a, b, c}, and P = {S → aSa, S → bSb, S → c}	4	2,3	6
17	Convert the PDA P= ({p, q},{0,1},{X,Z0}, δ, q, Z0) to a CFG. Where δ is given below: δ (q0, 0, S) = {(q0 , AS)} δ (q0, 0, A) = {(q0, AA), (q1, S)} δ (q0, 1, A) = {(q1, ε)} δ (q1, 1, A) = {(q1, ε)} δ (q1, ε, A) = {(q1, ε)} δ (q1, ε, S) = {(q1, ε)}	4	2,3	5

18	Convert the PDA P= ({p, q},{0,1},{X,Z0}, δ, q, Z0) to a CFG , if is given by δ(q, 1, Z0) ={(q, XZ0)} δ(q, 1, X) = {(q, XX)} δ(q, 0, X) = {(p, X)} δ(q, ε, X) = {(q, ε)} δ(p, 1, X) = {(p, ε)} δ(p, 0, Z0) = {(q, Z0)}	4	2,3	5

Unit – V

Sl. No	Questions	CO	PO	BT
Part – A				
1	What is a Turning Machine?	5	1	4
2	Define a Turing Machine.	5	1	1
3	Define Instantaneous description of TM.	5	1	1
4	What are the applications of TM?	5	1	4
5	What are the required fields of an instantaneous description or configuration of a TM.	5	1	4
6	Differentiate PDA and TM.	5	2	3
7	Define Universal TM	5	1	1
8	When is a function f said to be Turing computable?	5	2	4
9	Explain the Class of Grammars.	5	1	2
10	Discuss about PCP.	5	2	2
11	Differentiate PCP and MPCP.	5	2	4
Part – B				
12	Design a TM to recognize the language L ={$a^n b^n$; n>0} and test whether the strings “aabb” is accepts or not.	5	2,3	6
13	Design a TM to recognize the language L ={ww^r ; w ∈ (a+b)*} and test whether the strings “abba” is accepts or not.	5	2,3	6
14	Design a TM to recognize the language L ={wcw^r ; w ∈ (0+1)*}.	5	2,3	6
15	Design a Turing machine to compute proper subtraction m-n.	5	2,3	6
16	Explain the class of Grammars with example.	5	1,2	2
17	Explain the PCP and MPCP with example	5	1,2	2