INSTITUTE OF AERONAUTICAL ENGINEERING

DUNDIGAL – 500 043, HYDERABAD

COMPUTER SCIENCE AND ENGINEERING TUTORIAL QUESTION BANK

Course Name	:	**FORMAL LANGUAGES AND AUTOMATA THEORY**
Course Code	:	A40509
Class	:	II B. Tech II Semester
Branch	:	Computer Science and Engineering
Year	:	2016 – 2017
Course Faculty	:	Dr K Rajendra Prasad , Professor Ms N Mamatha, Assistant Professor Ms M Sandhya Rani, Assistant Professor Ms T Ramya, Assistant Professor

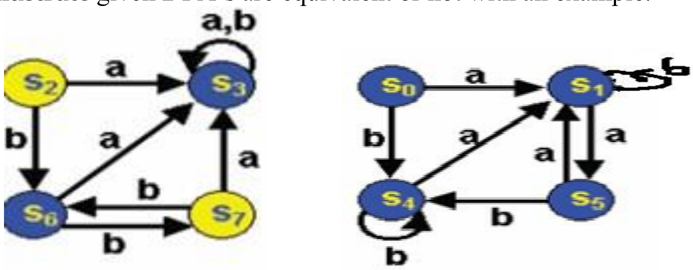
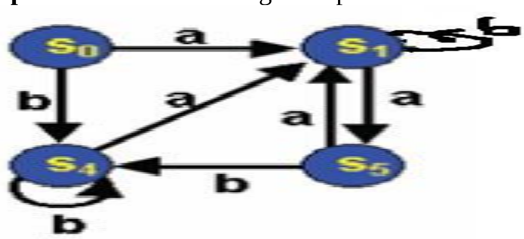
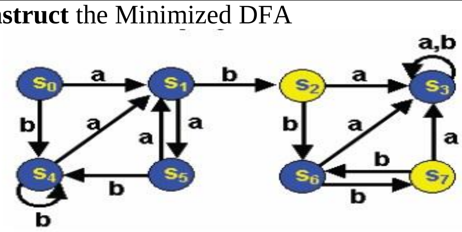
OBJECTIVES

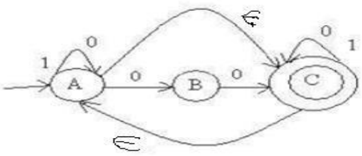
To meet the challenge of ensuring excellence in engineering education, the issue of quality needs to be addressed, debated and taken forward in a systematic manner. Accreditation is the principal means of quality assurance in higher education. The major emphasis of accreditation process is to measure the outcomes of the program that is being accredited.

In line with this, Faculty of Institute of Aeronautical Engineering, Hyderabad has taken a lead in incorporating philosophy of outcome based education in the process of problem solving and career development. So, all students of the institute should understand the depth and approach of course to be taught through this question bank, which will enhance learner's learning process.

Group - A (Short Answer Questions)

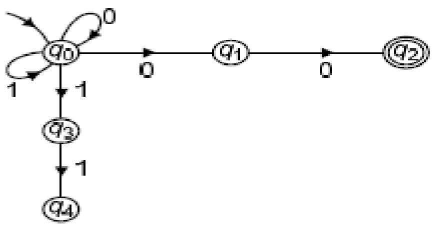
S. No.	Questions	Blooms Taxonomy Level	Course Outcomes
	UNIT - I		
	Part- A (Short Answer Questions)		
1.	**Explain** transition diagram, transition table with example.	Understand	1
2.	**Define** transition function of DFA.	Remember	2
3.	**Define** ε –transitions.	Remember	2
4.	**Construct** a DFA to accept even number of 0's.	Apply	2
5.	**Define** Kleene closure and positive closure.	Remember	1
6.	**Construct** a DFA to accept empty language.	Apply	2
7.	**Explain** power of an alphabet (∑*)?	Understand	1
8.	**Write** transition diagram for DFA accepting string ending with 00 defined over an alphabet ∑= {0,1}	Apply	2
9.	**Write** transition diagram for DFA to accept exactly one a defined over an alphabet ∑= {a,b}	Apply	2
10.	**Define** NFA with an example.	Remember	2
11.	**Explain** the different Operations on the languages.	Understand	
12.	**Construct** a finite automaton accepting all strings over {0, 1} having even number of 0's	Apply	2

13.	**Define** Moore Machines.	Remember	3
14.	**Define** Mealy Machines.	Remember	3
15.	**Write** DFA for odd number of 1's.	Apply	2
16.	**Write** NFA for (0+1)*101(0+1)*.	Apply	2
17.	**Write** DFA for (0+1)*10(0+1)*.	Apply	2
18.	**Define** ε - closure.	Remember	2
19.	**Write** NFA for (0+1)*001(0+1)*.	Apply	2
20.	**Write** DFA for (0+1)*00(0+1)*.	Apply	2
21	**Define** FSM and its structure with an example**.**	Remember	2
22	**Give** any two comparisions between NFA and DFA	Remember	2
Part- B (Long Answer Questions)			
1.	**Construct** a DFA to accept set of all strings ending with 010. Define language over an alphabet ∑ = { 0,1} and write for the above DFA .	Apply	2
2.	**Construct** a Moore machine to accept the following language. L = { w |w mod 3 = 0} on ∑ = { 0,1,2}	Apply	3
3.	**Write** any six differences between DFA and NFA	Apply	2
4.	**Write** NFA with Ɛ to NFA conversion with an example.	Understand	2
5.	**Construct** NFA for (0 + 1)*(00 + 11)(0 + 1)* and Convert to DFA.	Apply	2
6.	**Design** DFA for the following languages shown below ∑ = { a,b} a) L={w/ w does not contain the substring ab} b) L={w/ w contains neither the substring ab nor ba} c) L={w/ w is any string that doesn't contain exactly two a} d) L={w/ w is any string except a and b}	Apply	2
7.	**Illustrate** given 2 FA's are equivalent or not with an example. 	Apply	6
8.	**Construct** Mealy machine for (0 + 1)*(00 + 11) and convert to Moore machine.	Apply	3
9.	**Convert** NFA with Ɛ – a*b* to NFA.	Understand	2
10.	**Construct** NFA for (0 + 1)*101 and Convert to DFA.	Apply	2
11.	**Construct** a mealy machine that takes binary number as input and produces 2's complement of that number as output.Assume the string is read LSB to MSB and end carry is discarded.	Understand	3
12.	**Explain** with the following example the Minimize the DFA . 	Understand	2
13.	**Construct** a DFA, the language recognized by the Automaton being $L = \{a^n b / n \geq 0\}$. Draw the transition table.	Apply	2
14.	**Construct** the Minimized DFA 	Apply	2
15.	**Construct** the DFA that accepts/recognizes the language $L(M) = \|$ $w \in \{a, b, c\}*$ and w contains the pattern *abac* }. Draw the	Apply	2

	transition table.		
16.	**Construct** NFA for given NFA with Є-moves 	Apply	2
17.	**Differentiate** between DFA and NFA with an example.	Understand	2
18.	**Construct** a finite automaton accepting all strings over {0, 1} having even number of 0’s and even number of 1’s.	Apply	2
19.	**Construct** a Moore Machine to determine the residue mod 5 for each binary string treated as integer. Sketch the transition table.	Apply	3
20.	**Construct** the Moore Machine for the given Mealy machine	Understand	3

STATE/I	a	b	output
q0	q1	q2	1
q1	q1	q1	0
q2	q1	q0	1

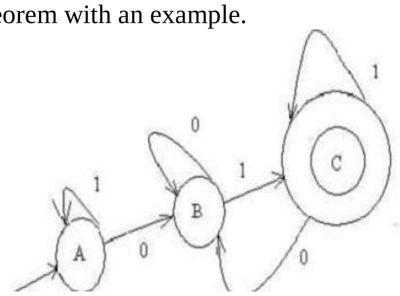
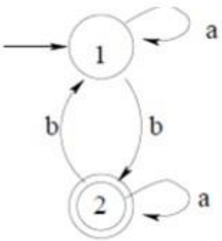
Part- C (Problem Solving and Critical Thinking)

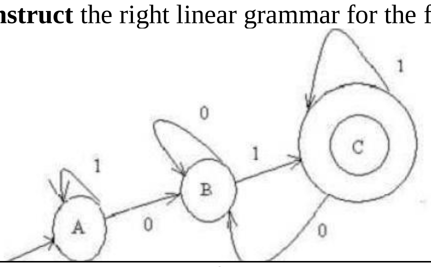
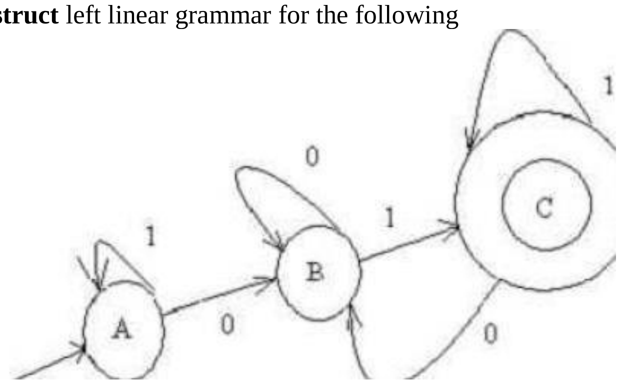
1	**Construct** NFA for (0 + 1)*0(0 + 1)0(0 + 1)* and convert to DFA.	Apply	2
2	**Construct** NFA for (0 + 1)*010(0 + 1)* and Convert to DFA.	Apply	2
3	**Construct** NFA with Ɛ for 0*1*12* and Convert to NFA .	Apply	2
4	**Construct** Mealy Machine for Residue Modulo of 5 for the ternary number system and convert to Moore Machines.	Apply	2
5	**Write** the DFA that will accept those words from $\Sigma = \{a, b\}$ where the number of a’s is divisible by two and the number of *b*’s is divisible by three. Sketch the transition table of the finite	Apply	2
6	**Construct** DFA for the given NFA as shown in fig. below 	Apply	2

UNIT – II

Part- A (Short Answer Questions)

1.	**Define** Regular Languages.	Remember	7
2.	**Define** Pumping Lemma for Regular Languages.	Remember	7
3.	**Write** the applications of pumping lemma for regular languages.	Apply	7
4.	**List** any two applications of regular expression.	Remember	7
5.	**Define** Context Free Grammars.	Remember	8
6.	**Define** Left linear derivation.	Remember	7
7.	**Write** regular expression for denoting language containing empty string.	Apply	7
8.	**Differentiate** left linear and right linear derivations.	Understand	8
9.	**Write** the Context free grammar for palindrome.	Remember	8
10.	**Define** right linear grammars.	Remember	7
11.	**Define** Regular grammars.	Remember	7
12.	**Write** regular expressions for the Set of strings over {0, 1} whose last two symbols are the same.	Apply	7
13.	**Define** right linear derivation.	Remember	7
14.	**Define** left linear grammars.	Remember	7
15.	**Write** the regular language generated by regular expression (0+1)*001(0+1)*.	Apply	7
16.	**Write the** Regular Expression for the set of binary strings.	Apply	7
17.	**Write** the derivation of the string aaaa from CFG –	Apply	8

	S → a S/A A → a		
18.	**Write** the derivation of the string 110 from CFG – S→ A0/B A→0/12/B B →A/11	Apply	8
19.	**Write** the Regular Expression to generate atleast one b over Σ ={a,b}	Apply	8
20.	**Write** the Context free grammar for equal number of a's and b's.	Apply	8
Part- B (Long Answer Questions)			
1.	**Convert** Regular Expression 01* + 1 to Finite Automata.	Understand	7
2.	**Convert** given Finite Automata to Regular Expression using Arden's theorem with an example. 	Understand	7
3.	**Construct** Right linear , Left linear Regular Grammars for 01*+1.	Apply	7
4.	**Explain** Identity rules . Simplify the Regular Expression - Є + 1*(011)*(1*(011)*)*	Understand	7
5.	**Construct** Regular grammar for the given Finite Automata. (a+b)*ab*.	Apply	7
6.	**Construct** Leftmost Derivation. , Rightmost Derivation, Derivation Tree for the following grammar S→ aB | bA A→ a |aS |bAA B→ b | bS |aBB For the string aaabbabbba .	Apply	8
7.	**Explain** the properties, applications of Context Free Languages	Understand	8
8.	**Construct** right linear and left linear grammars for given Regular Expression.	Apply	7
9.	**Construct** a Transition System M accepting L(G) for a given Regular Grammar G.	Apply	7
10.	**Discuss** the properties of Context free Language. Explain the pumping lemma with an example.	Understand	7
11.	**Write** regular expressions for the given Finite Automata 	Apply	7
12.	**Construct** a NFA with Є equivalent to the regular expression 10 + (0 + 11)0*1	Apply	7
13.	**Construct** Leftmost Derivation. , Rightmost Derivation, Derivation Tree for the following grammar G = (V, T, P, S) with N = {E}, S = E, T = {id, +, *(,)} E→ E+E E→E* E E→ (E) E→ id Obtain id+id*id in right most derivation, left most derivation	Apply	7
14.	**Write** a CFG that generates equal number of a's and b's.	Apply	8
15.	**Convert** G = ({S},{a},{ S → aS /a},{S}) into FA	Understand	7
16.	**Construct** a Regular expression for the set all strings of 0's and 1's	Apply	7

	with at least two consecutive 0's		
17.	**Construct** context free grammar which generates palindrome strings $\sum=\{a,b\}$	Apply	8
18.	**Construct** equivalent NFA with ϵ for the given regular expression 0*(1(0+1))*.	Apply	7
19.	**Construct** the right linear grammar for the following 	Apply	7
20.	**Write** 12 identity rules for regular expressions	Apply	7
Part- C (Problem Solving and Critical Thinking)			
1	**Convert** Regular Expression (11 + 0)*(00 + 1)* to NFA with Ɛ.	Understand	7
2	**Convert** Regular Expression (a + b)*(aa + bb)(a + b)* to DFA.	Understand	7
3	**Construct** Regular Grammars for Finite Automata 0*(1(0 + 1))* .	Apply	7
4	**Construct** Finite Automata for A0 → a A1 A1 →	Apply	7
5	**Construct** left linear grammar for the following 	Apply	7
UNIT – III			
Part- A (Short Answer Questions)			
1.	**Define** Greibach normal form.	Remember	9
2.	**Define** nullable Variable.	Remember	8
3.	**Write** the minimized CFG for the following grammar S→ABCa | bD A→BC |b B→b | ε C→Đ | ε D→d	Remember	9
4.	**Convert** the grammar to CNF - S →bA/aB A→aS/a B→bS/b.	Understand	8
5.	**Explain** the elimination of UNIT production.	Understand	8
6.	**Explain** the elimination of useless symbols in productions.	Understand	8
7.	**Define** CNF.	Remember	9
8.	**Write** the minimization of CFG – S → a S/A A →a B → aa	Understand	8
9.	**Define** the ambiguity in CFG.	Remember	8
10.	**What** is the use of CNF and GNF.		8
11.	**Write** the minimization of CFG - S → aS1b S1→aS1b/ε.	Understand	8
12.	**Write** the minimization of CFG - S → A A →aA/ ε.	Understand	8
13.	**Write** the minimization of CFG - S →AB / a A → a.	Understand	8
14.	**Write** the minimization of CFG - S→aS/A/C A →a B →aa C →aCb.	Understand	8
15.	**Write** the minimization of CFG - S→AbA A→Aa/ ε.	Understand	8

16.	**Write** the minimization of CFG - S→aSa S →bSb S→a/b/ ε.	Understand	8
17.	**Write** the minimization of CFG - S→ A0/B A→0/12/B B →A/11.	Understand	8
18.	**Convert** the grammar to CNF - S→aSa/aa S→bSb/bb S →a/b.	Understand	8
19.	**Convert** the grammar to CNF - S→aAbB A→aA/a B→bB/a.	Understand	8
20.	**Define** PDA.	Remember	10
21.	**Define** NPDA.	Remember	10
22.	**Differentiate** between deterministic and nondeterministic PDA.	Understand	10
23.	**Define** the language of DPDA.	Remember	10
24.	**List** the steps to convert CFG to PDA.	Remember	11
25.	**Explain** – acceptance of PDF by final state.	Understand	10
26.	**Explain** – acceptance of PDF by empty stack.	Understand	10
27.	**Convert** the following PDA to CFG δ(q0,b,z0)={q0,zz0)	Apply	11
28.	**Convert** the following PDA to CFG δ(q0, b, z)=(q0,zz)	Apply	11
29.	**Convert** the following PDA to CFG δ(q0, ϵ ,z0)=(q0,ϵ)	Apply	11
30.	**Convert** the following PDA to CFG δ(q0,a,z) = (q1,z)	Apply	11
31.	**Convert** the following PDA to CFG δ(q1,b,z)=(q1,ϵ)	Apply	11
32.	**Convert** the following PDA to CFG δ(q1,a,z0)=(q0,z0)	Apply	11
33.	**Convert** the following PDA to CFG δ(q0,0,z0)={q0,xz0)	Apply	11
34.	**Convert** the following PDA to CFG δ(q0,0,x)=(q0,xx)	Apply	11
35.	**Convert** the following PDA to CFG δ(q0,1,x)=(q1,ϵ)	Apply	11
36.	**Convert** the following PDA to CFG δ(q1,1,x) = (q1,ϵ)	Apply	11
37.	**Convert** the following PDA to CFG δ(q1,ϵ,x)=(q1,ϵ)	Apply	11
38.	**Convert** the following PDA to CFG δ(q1,ϵ,z0)=(q1,ϵ)	Apply	11
39.	**Convert** the following PDA to CFG δ(q1,ϵ,z)=(q0,ϵ)	Apply	11
40.	**Convert** the following CFG to PDA S□ ABC | BbB	Apply	11
41.	**Convert** the following CFG to PDA A□ aA | BaC|aaa	Apply	11
42.	**Convert** the following CFG to PDA B□ bBb| a|D	Apply	11
43.	**Convert** the following CFG to PDA C□CA|AC	Apply	11
44.	**Convert** the following CFG to PDA S □ a S/A	Apply	11
Part- B (Long Answer Questions)			
1.	**Write** a short notes on Chomsky Normal Form and Griebach Normal Form.	Apply	9
2.	**Show** that the following grammar is ambiguous with respect to the string aaabbabbba. S→ aB | bA A→ aS| bAA|a B→ bS | aBB | b	Understand	8
3.	**Use** the following grammar : S→ ABC | BbB A→ aA |BaC|aaa B→ bBb| a|D C→CA|AC D→ **ε** Eliminate **ε**-productions. Eliminate any unit productions in the resulting grammar. Eliminate any useless symbols in the resulting grammar. Convert the resulting grammar into Chomsky Normal Form	Apply	9
4.	**Illustrate** the construction of Griebach normal form with an example.	Apply	9

5.	**Show** that the following CFG ambiguous. S→ iCtS | iCtSeS | a C→ b	Apply	8
6.	**Discuss** the Pumping lemma for Context Free Languages concept with example $\{a^n b^n c^n$ where n>=0}	Understand	9
7.	**Write** the simplified CFG productions in S → a S1b S1 → a S1b/ Є	Apply	8
8.	**Convert** the following CFG into GNF. S→AA/a A → SS/b	Understand	8
9.	**Explain** unit production? Explain the procedure to eliminate unit production.	Understand	8
10.	**Explain** the procedure to eliminate ϵ-productions in grammar.	Understand	8
11.	**Convert** the following grammar into GNF G=({A1,A2,A3},{a,b},P,A) A1->A2A3 A2->A3A1/b A3->A1A2/a	Understand	8
12.	**Write** simplified CFG productions from the following grammar A->aBb/bBa B->aB/bB/ϵ	Apply	8
13.	**Convert** the following grammar into GNF S->ABA/AB/BA/AA/B A->aA/a B->bB/b	Understand	8
Part- C (Problem Solving and Critical Thinking)			
1	**Construct** PDA for equal number of x's and y's	Apply	10
2	**Convert** the following grammar into GNF A1 →A2 A3 A2→ A3 A1 /b A3→ A1 A2 /a	Understand	9
3	**Construct** DPDA for L = { $W\#W^R$ /W ϵ (X + Y)*}	Apply	10
4	**Convert the following PDA to CFG** **δ(q0,0,z0)={q0,xz0)** **δ(q0,0,x)=(q0,xx)** **δ(q0,1,x)=(q1,ϵ)** **δ(q1,1,x) = (q1,ϵ)** **δ(q1,ϵ,x)=(q1,ϵ)** **δ(q1,ϵ,z0)=(q1,ϵ)**	Understand and	11
5	**Write the PDA that accepts the language{a^m b^n/n>m}**	Apply	10
6	**Design a PDA for the following grammar** **S->0A** **A->0AB/1** **B->1**	Create	10
7	**Convert the following PDA to CFG** **M=({q0,q1},{a,b},{z0,za},µ,q0,z0,Φ)** **δ is given by, δ(q0,a,z0)=(q0,zz)** **δ(q0,a,z)=(q0,zz0)** **δ(q0,b,z)=(q1,ϵ)** **δ(q1,b,z)=(q1,ϵ)** **δ(q1,ϵ,z0)=(q1,ϵ)**	Understand and	11
UNIT - IV			
Part- A (Short Answer Questions)			
1.	**Define** Turing Machine	Apply	12
2.	**Explain** the moves in Turing Machine.	Understand	12
3.	**Define** an Instantaneous Description of a Turing Machine.	Remember	12
4.	**Define** the Language of Turing Machine.	Remember	12
5.	**List** types of TM.	Remember	12
6.	**Define** Computable Functions by Turing Machines .	Remember	12
7.	**Write** the difference between Pushdown Automata and Turing	Apply	12

	Machine.		
8.	**Explain** Church's Hypothesis.	Understand	12
9.	**Define** Context sensitive language.	Remember	12
10.	**Define** multi head Turing Machine.	Remember	12
11.	**Define** multi dimensional Turing Machine.	Remember	12
12.	**Define** multiple tapes Turing Machine.	Remember	12
13.	**Define** Recursive languages.	Remember	12
14.	**Define** Recursively enumerable languages.	Remember	12
15.	**Define** Two way infinite Turing Machine.	Remember	12
16.	**Define** Non deterministic Turing Machine.	Remember	12
17.	**Define** Counter machine.	Remember	12
18.	**Explain** the model of Turing machine.	Remember	12
19.	**Construct** Turing Machine for 1's complement for binary numbers.	Remember	12
20.	**Differentiate** Recursive languages and Recursively enumerable languages.	Remember	12
Part- B **(Long Answer Questions)**			
1.	**Define** a Turing Machine. With a neat diagram explain the working of a Turing Machine.	Remember	12
2.	**Differentiate** Turing Machine with other automata.	Apply	12
3.	**Construct** a Transition diagram for Turing Machine to accept the following language. $L = \{ 0^n1^n0^n \mid n \geq 1\}$	Apply	12
4.	**Construct** Transition diagram for Turing Machine that accepts the language $L = \{0^n1^n \mid n \geq 1\}$. Give the transition diagram for the Turing Machine obtained and also show the moves made by the Turing machine for the string 000111.	Apply	12
5.	**Construct** a Transition diagram for Turing Machine to accept the language $L= \{ w\#w^R \mid w \in (a + b)^* \}$	Apply	12
6.	**Write** short notes on Recursive and Recursively Enumerable languages.	Apply	12
7.	**Write** the properties of recursive and recursively enumerable languages.	Apply	12
8.	**Construct** a Turing Machine to accept strings formed with 0 and 1 and having substring 000.	Apply	12
9.	**Construct** a Turing Machine that accepts the language $L = \{1^n2^n3^n \mid n \geq 1\}$. Give the transition diagram for the Turing Machine obtained and also show the moves made by the Turing machine for the string 111222333.	Apply	12
10.	**Define** Linear bounded automata and explain its model?	Apply	12
11.	Explain the power and limitations of Turing machine.	Create	12
12.	Construct Transition diagram for Turing Machine - $L=\{a^nb^nc^n/n>=1\}$	Apply	12
13.	Construct a Transition diagram for Turing Machine to implement addition of two unary numbers(X+Y).	Apply	12
14.	Construct a Linear Bounded automata for a language where $L=\{a^nb^n/n>=1\}$	Apply	12
15.	Explain the types of Turing machines.	Apply	12
16.	Write briefly about the following a)Church's Hypothesis b)Counter machine	Apply	12
17.	Construct a Transition table for Turing Machine to accept the following language. $L = \{ 0^n1^n0^n \mid n \geq 1\}$	Apply	12
18.	Construct a Transition diagram for Turing Machine to accept the language $L= \{ ww^R \mid w \in (a + b)^* \}$	Apply	12
19.	Construct Transition table for TM - $L=\{a^nb^nc^n/n>=1\}$	Apply	12
20.	Construct a Linear Bounded automata for a language where $L=\{a^nb^nc^n/n>=1\}$	Apply	12
Part- C (Problem Solving and Critical Thinking)			
1	Construct a Turing Machine that accepts the language $L = \{a^{2n}b^n \mid n \geq 0\}$. Give the transition diagram for the Turing Machine obtained.	Apply	12

<table>
<tr><td>2</td><td>Construct a Turing Machine that gives two’s compliment for the given binary representation.</td><td>Apply</td><td>12</td></tr>
<tr><td>3</td><td>Construct a Turing Machine to accept the following language. L = { $w^n x^n y^n z^n$ | n ≥1}</td><td>Apply</td><td>12</td></tr>
<tr><td colspan="4">UNIT - V</td></tr>
<tr><td colspan="4">Part- A (Short Answer Questions)</td></tr>
<tr><td>1.</td><td>Define Chomsky hierarchy of languages.</td><td>Knowledge</td><td>4</td></tr>
<tr><td>2.</td><td>Define Universal Turing Machine</td><td>Knowledge</td><td>12</td></tr>
<tr><td>3.</td><td>Define Context sensitive language.</td><td>Knowledge</td><td>5</td></tr>
<tr><td>4.</td><td>Define decidability.</td><td>Knowledge</td><td>13</td></tr>
<tr><td>5.</td><td>Define P problems.</td><td>Knowledge</td><td>13</td></tr>
<tr><td>6.</td><td>Define Universal Turing Machines</td><td>Knowledge</td><td>13</td></tr>
<tr><td>7.</td><td>Give examples for Undecidable Problems</td><td>Understand</td><td>13</td></tr>
<tr><td>8.</td><td>Define Turing Machine halting problem.</td><td>Knowledge</td><td>13</td></tr>
<tr><td>9.</td><td>Define Turing Reducibility</td><td>Knowledge</td><td>13</td></tr>
<tr><td>10.</td><td>Define Post’s Correspondence Problem.</td><td>Knowledge</td><td>13</td></tr>
<tr><td>11.</td><td>Define Type 0 grammars .</td><td>Knowledge</td><td>4</td></tr>
<tr><td>12.</td><td>Define Type 1 grammars .</td><td>Knowledge</td><td>4</td></tr>
<tr><td>13.</td><td>Define Type 2 grammars .</td><td>Knowledge</td><td>4</td></tr>
<tr><td>14.</td><td>Define Type 3 grammars .</td><td>Knowledge</td><td>4</td></tr>
<tr><td>15.</td><td>Define NP problems.</td><td>Knowledge</td><td>13</td></tr>
<tr><td>16.</td><td>Define NP complete problems</td><td>Knowledge</td><td>13</td></tr>
<tr><td>17.</td><td>Define NP Hard problems</td><td>Knowledge</td><td>13</td></tr>
<tr><td>18.</td><td>Define undecidability problem.</td><td>Knowledge</td><td>13</td></tr>
<tr><td>19.</td><td>Define turing Reducibility.</td><td>Knowledge</td><td>13</td></tr>
<tr><td>20.</td><td>List the types of grammars.</td><td>Knowledge</td><td>13</td></tr>
<tr><td colspan="4">Part- B (Long Answer Questions)</td></tr>
<tr><td>1.</td><td>Explain the concept of decidable and undecidability problems about Turing Machines.</td><td>Understand</td><td>12</td></tr>
<tr><td>2.</td><td>Write briefly about Chomsky hierarchy of languages..</td><td>Apply</td><td>13</td></tr>
<tr><td>3.</td><td>Explain individually classes P and NP</td><td>Understand</td><td>13</td></tr>
<tr><td>4.</td><td>Write a shot notes on post's correspondence problem and check the following is PCP or not.<table><tr><td>I</td><td>A</td><td>B</td></tr><tr><td>1</td><td>11</td><td>111</td></tr><tr><td>2</td><td>100</td><td>001</td></tr><tr><td>3</td><td>111</td><td>11</td></tr></table></td><td>Apply</td><td>13</td></tr>
<tr><td>5.</td><td>Explain the Halting problem and Turing Reducibility.</td><td>Understand</td><td>13</td></tr>
<tr><td>6.</td><td>Write a short notes on universal Turing machine.</td><td>Apply</td><td>12</td></tr>
<tr><td>7.</td><td>Write a short notes on Chomsky hierarchy.</td><td>Apply</td><td>4</td></tr>
<tr><td>8.</td><td>Write a short notes on Context sensitive language and linear bounded automata.</td><td>Apply</td><td>4</td></tr>
<tr><td>9.</td><td>Write a short note on NP complete</td><td>Apply</td><td>13</td></tr>
<tr><td>10.</td><td>Write a short note on NP hard problems.</td><td>Apply</td><td>13</td></tr>
<tr><td>11.</td><td>Write a shot notes on post's correspondence problem and check the following is PCP or not.<table><tr><td>I</td><td>A</td><td>B</td></tr><tr><td>1</td><td>100</td><td>1</td></tr><tr><td>2</td><td>0</td><td>100</td></tr><tr><td>3</td><td>1</td><td>0</td></tr></table></td><td>Apply</td><td>13</td></tr>
<tr><td>12.</td><td>Write a shot notes on post's correspondence problem and check the following is PCP or not.<table><tr><td>I</td><td>A</td><td>B</td></tr><tr><td>1</td><td>00</td><td>0</td></tr><tr><td>2</td><td>001</td><td>11</td></tr></table></td><td>Apply</td><td>13</td></tr>
</table>

	3	1000	011		
UNIT - V			
1	**Explain** PCP and MPCP with examples.	Understand	13
2	**Explain** Turing theorem ,Halting problems, Turing Reducibility.	Understand	13
3	**Explain** Type 3 and Type 2 grammars with example.	Apply	4
4	**Explain** Type 1 and Type 0 grammars with example.	Apply	4