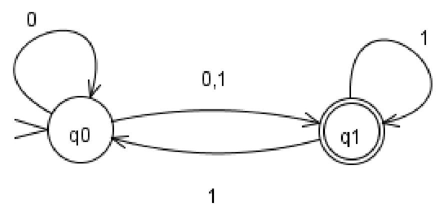
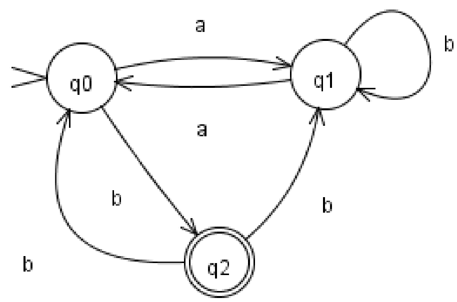
Adi Shankara Institute of Engineering and Technology
Kalady, Ernakulam, Kerala, India - 683574
Department of
Computer Science & Engineering and Information Technology

CS301 – THEORY OF COMPUTATION
QUESTION BANK

MODULE 1

Sl No.	Questions	Knowledge Level
Introduction to Automata Theory and its significance. Type 3 Formalism: Finite state automata – Properties of transition functions, Designing finite automata, NFA, Finite Automata with Epsilon Transitions, Equivalence of NFA and DFA, Conversion of NFA to DFA, Equivalence and Conversion of NFA with and without Epsilon Transitions. (CO2)		
Sl No.	**Questions**	**Knowledge Level**
1.	Why are switching circuits called as finite state systems?	**K2**
2.	Conversion of NFA to DFA a. Draw the NFA's transition table. b. Take the initial state of NFA be the initial state of DFA. c. Transit the initial state for all the input symbols. d. If new state appears transit it again and again to make all state as old state. All the new states are the states of the required DFA' e. Draw the table for DFA. f. Draw the DFA from the transition table.	**K3**
3.	Explain symbols, alphabets, strings with examples.	**K2**
4.	Write down the quad tuple definition of formal grammar.	**K1**
5.	Find the grammar for the language $a^n b^n / n >= 0$.	**K1**
6.	Define automata. What are the two types of automata.	**K1**
7.	What are the components of a Finite Automata.	**K1**
8.	Write formal definition of Deterministic Finite Automata (DFA).	**K1**
9.	Explain language acceptability of DFA.	**K2**
10.	Explain extended transition function.	**K2**
11.	Write formal definition of NFA.	**K1**
12.	Explain language acceptability of NFA.	**K2**
13.	Explain extended transition function.	**K2**
14.	Write formal definition of NFA-Є.	**K1**
15.	Define Є-closure.	**K1**

<table>
<tr><td>16.</td><td>What is meant by dead state of DFA. Explain with an example.</td><td>K1</td></tr>
<tr><td>17.</td><td>What is meant by indistinguishable state of DFA. Explain with an example.</td><td>K1</td></tr>
<tr><td>18.</td><td>Differentiate NFA and DFA.</td><td>K2</td></tr>
<tr><td>19.</td><td>Construct a DFA corresponding to the regular expression 1(1+0)*.</td><td>K3</td></tr>
<tr><td>20.</td><td>Find DFA for the language L= {$a^n b^m$, n>=2; m>=1}</td><td>K3</td></tr>
<tr><td>21.</td><td>Design a DFA that accepts the language: L= {w/w ends with 01} form alphabet {0,1}.</td><td>K3</td></tr>
<tr><td>22.</td><td>Design a DFA that accepts odd number of a’s and even number of b’s over {a,b}.</td><td>K3</td></tr>
<tr><td>23.</td><td>Design a DFA to accept the language L={w / n_b(w) mod 3 >1} over ∑={a,b}.</td><td>K3</td></tr>
<tr><td>24.</td><td>Design an NFA to accept set of strings with at least two consecutive 0’s or two consecutive 1’s.</td><td>K3</td></tr>
<tr><td>25.</td><td>Design an NFA for the language L= {$abab^n$:n>=0} U {aba^n: n>=0} with not more than 5 states.</td><td>K3</td></tr>
<tr><td>26.</td><td>Find Є-closure for each state in the given NFA.
<table>
<tr><td>State\Symbol</td><td>Є</td><td>A</td><td>B</td><td>C</td></tr>
<tr><td>->p</td><td>{}</td><td>{p}</td><td>{q}</td><td>{r}</td></tr>
<tr><td>Q</td><td>{p}</td><td>{q}</td><td>{r}</td><td>{}</td></tr>
<tr><td>*r</td><td>{q}</td><td>{r}</td><td>{}</td><td>{p}</td></tr>
</table>
</td><td>K3</td></tr>
<tr><td>27.</td><td>Prove the equivalence of NFA and DFA.</td><td>K2</td></tr>
<tr><td>28.</td><td>Explain the conversion of NFA to DFA with an example.</td><td>K2</td></tr>
<tr><td>29.</td><td>Design an NFA that accept the language L={w/w contains substring 00} from the input alphabet {0,1}. Convert that NFA to DFA by the method of subset construction.</td><td>K2</td></tr>
<tr><td>30.</td><td>List some of the applications of automata theory. (3Marks)(CS(S), April 2018)</td><td>K1</td></tr>
</table>

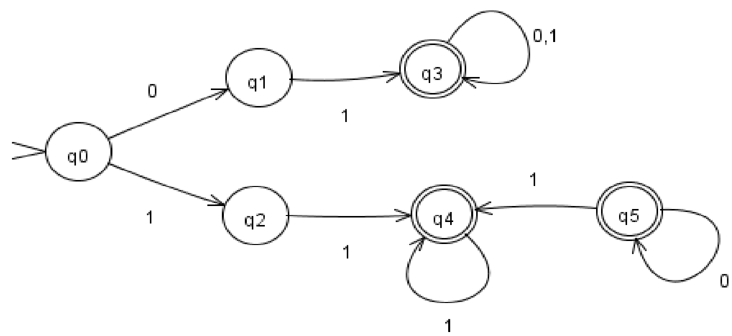
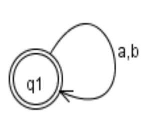
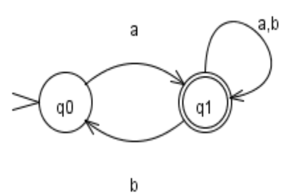
31.	Convert the given NFA to DFA  	**K2**
32.	Convert the given NFA to DFA  	**K2**
33.	Define Non Deterministic Finite Automata? Compare its ability with Deterministic Finite Automata in accepting languages. (3 Marks)(CS(R), Dec 2017)	**K1**
34.	Write the notations for the language accepted by DFA, NFA, ε -NFA. (3Marks)(CS(R), Dec 2017)	**K2**
35.	a) Design a Finite state automata which accepts all strings over {0,1} with odd number of l's and even number of 0's. (5Marks) b) Show the changes needed to convert the above designed automata to accept even number of l's and odd number of 0's (4Marks) (CS(R), Dec 2017)	**K3**
36.	Construct regular expression for the language that consists of all strings ending with 00. Assume Σ= {0, 1}. (3Marks)(CS(S), April 2018)	**K3**
37.	Design non deterministic automata (without ε moves) for the regular language that consist of all strings with at least two consecutive 0's. Assume Σ = {0, 1}. (3Marks)(CS(S), April 2018)	**K3**
38.	Prove the equivalence of non deterministic finite automata and deterministic finite automata. (9Marks)(CS(S), April 2018)	**K2**
39.	a) Design an NFA for L={w|w has at least 2 consecutive 0's or 1's over ∑={0,1}}. (6Marks) b) Define the language of DFA, NFA and NFA-ε. (5Marks)	**K3**

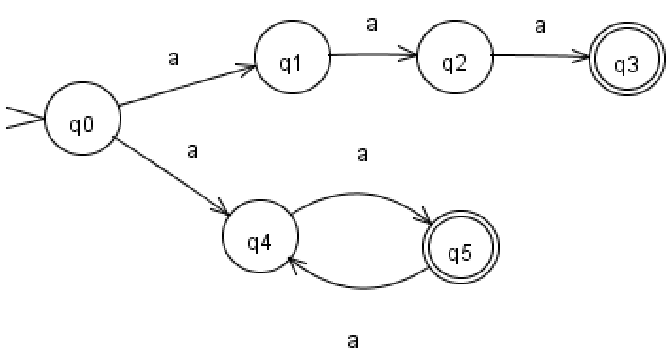
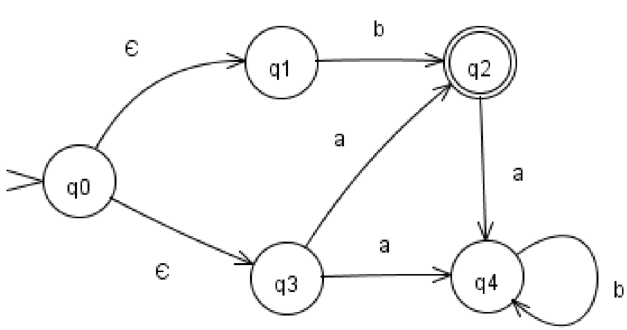
<table>
<tr><td></td><td>c) Convert the following NFA to DFA.
<table>
<tr><th>δ</th><th>0</th><th>1</th></tr>
<tr><td>→ p</td><td>{p,q}</td><td>{p}</td></tr>
<tr><td>q</td><td>ϕ</td><td>{r}</td></tr>
<tr><td>*r</td><td>{p,r}</td><td>{q}</td></tr>
</table>
(4Marks)(IT(S), April 2018)</td><td></td></tr>
<tr><td>40.</td><td>a) Explain Chomsky classification of grammars. (5Marks)
b) If ∑={a,b,c} then write ∑1, ∑2, ∑3,∑*. (4Marks)
c) Show how an NFA can be created which accepts the reverse of a language. (6Marks)(IT(S), 2017)</td><td>K2</td></tr>
</table>

MODULE 2

Myhill-Nerode Theorem, Minimal State FA Computation. Finite State Machines with Output- Mealy and Moore machine (Design Only), Two- Way Finite Automata. Regular Grammar, Regular Expressions, Equivalence of regular expressions and NFA with epsilon transitions. Converting Regular Expressions to NFA with epsilon transitions Equivalence of DFA and regular expressions, converting DFA to Regular Expressions. (CO1, CO2)

Sl No.	Questions	Knowledge Level
1.	Find the grammar for generating set of all strings which are palindromes over {a,b} and show the derivation of the string ‘ababa’.	**K3**
2.	Find the grammar for generating set of all strings with equal number of a’s and b’s over {a,b} and show the derivation of the string ‘aababb’.	**K3**
3.	Write the grammar for the following languages: a) To generate strings with exactly one ‘a’ over {a,b}. b) To generate strings with atleast one ‘a’ over {a,b}. c) L= {$(ab)^n$ / n>0} over {a,b}. d) L={ $a^nb^nc^n$ / n>=0} over {a,b}. e) L={a^ib^j / i>j} over {a,b}.	**K1**
4.	What is the language generated by the G=({S}, {0,1}, P,S) where P is S-> 0S0/1S1/0/1?	**K1**
5.	Given a G =({S}, {a,b}, P, S) with P as S->aSb/Є. Obtain the language L(G) and a sentence in the language.	**K2**
6.	Given G with P as S-> AB, A->Aa/a, B->Bb/b. Show that w=aabbbb Є L(G).	**K2**
7.	Construct regular expression for generating even number of a’s followed by odd number of b’s.	**K3**

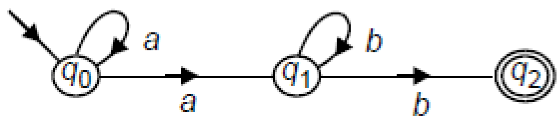
8.	Define regular expressions.	**K1**
9.	Construct regular expression for the language that have the set of strings over {a,b,c} containing at least one a and at least one b.	**K3**
10.	Construct regular expression for the language that have the set of all strings of 0's and 1's whose 10th symbol from the right end is 1.	**K3**
11.	Explain various notations used for presenting finite automata with example.	**K1**
12.	Explain the algorithm for minimization of DFAs with an example.	**K1**
13.	Minimize the given DFA.  	**K2**
14.	State and prove pumping lemma for regular languages. Prove that L = {a^p |p is a prime number} is not a regular language.	**K2**
15.	Explain any three applications of finite automata.	**K2**
16.	Design DFA for the following requirements: a) Even number of a's and b's. b) (0+1)*(1+00) c) To accept decimal numbers divisible by 3 d) All strings that have neither 00 or 11 as substrings. e) All strings that don't have the substring 110 f) Strings whose length is divisible by 4 g) L ={w/w Є{a,b,c}* and w contains the pattern abac}	**K3**
17.	Determine the language of the given DFA: a)  b)  	**K2**

18.	Obtain the DFA state diagram of M=(Q,Σ,δ, q0, F) where Q={q0,q1,q2,q3}, Σ={a,b}, q0 is initial and final state. δ(q0,a)=q2 δ(q3,a)=q1 δ(q2,b)=q3 δ(q1,a)=q3 δ(q0,b)=q1 δ(q3,b)=q2 δ(q2,a)=q0 δ(q1,b)=q0	**K3**
19.	Design NFA for the following requirements: I. L={w/w contains the substring 0101} with 5 states II. L={w/w contains at least two 0’s or exactly two 1’s} with 6 states. III. Strings over the alphabet containing either ‘cat’ or ‘rat.	**K3**
20.	Given L is the language accepted by NFA in the figure. Determine an NFA that accepts L U {a^5}. 	**K3**
21.	Design NFA-Є for the following requirements: I. L={a* U b*} II. To accept ‘abac’ and all its suffixes. III. To accept all strings with any number of a’s followed by any number of b’s followed by any number of c’s.	**K3**
22.	Find Є-closure of each state of the given FA: 	**K3**

23. Convert NFA to DFA — **K3**

Q\Σ	A	B
->q0	{q0, q1}	{q2}
Q1	{q0}	{q1}
*q2	{}	{q0,q1}

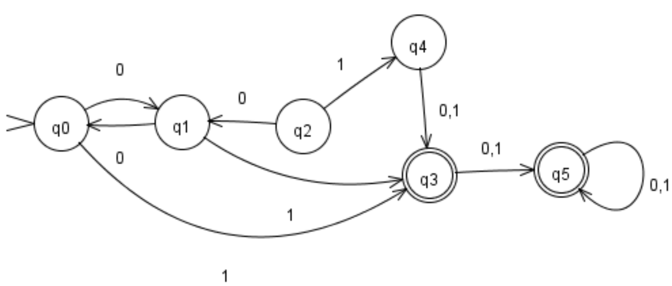
24. Convert NFA to DFA — **K3**



25. Convert NFA to DFA — **K3**

	0	1
p	{p,q}	P
q	r	R
r	s	-
s	s	S

26. Minimize the DFA. — **K2**



27. Construct a DFA for r= (a+b)*abb and minimize it. — **K3**

28. Write regular expressions for the following requirements: — **K1**

i) All strings over {a,b} containing a substring 'aba'.

ii) Set of strings over {a,b} having even number of characters.

iii) Set of all three lettered words, starting with b over {a,b}.

iv) Set of all strings with at least two b's over {a,b}.

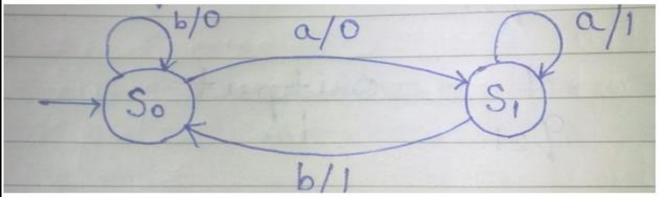
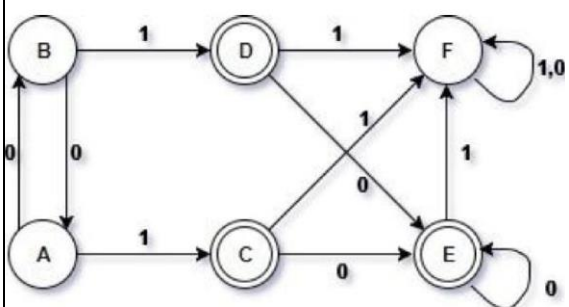
	v) Set of all strings that end in double letter over {a,b}. vi) L={w: |w|mod 3=0} over {a,b}	
29.	Show that (0*1*)* = (0+1)*	**K2**
30.	Construct NFA corresponding to given RE: (a) (a+b)*abb (b) 00*+1 (c) (0+1)*00(0+1)*	**K3**
31.	Explain the process of building finite automata from a regular expression.	**K2**
32.	Design Moore machine that gives an output of 1 if the input string ends in bab.	**K3**
33.	Design Moore machine to recognize the substring 1101.	**K3**
34.	Design Mealy machine that recognize double occurrence of symbol 'a'in input string S ∈ ∑* where ∑={a,b}	**K3**
35.	Convert Moore machine into Mealy machine. Moore machine given as:	**K3**
36.	Construct Mealy machine equivalent to the Moore Machine equivalent:	**K3**

Table for Q35:

Present State	Next State		Output
	0	1	
➔ q0	q3	q1	0
q1	q1	q2	1
q2	q2	q2	0
q3	q3	q3	0

Table for Q36:

Present State	Next State		Output
	a=0	a=1	
➔ q1	q4	q2	0
q2	q2	q3	1
q3	q3	q4	0
q4	q4	q1	0

37.	Convert the Mealy Machine to Moore Machine 	**K3**
38.	Draw a FA that accepts strings containing exactly 1 over alphabet {0, 1} and write a regular expression for the same.	**K2**
39.	Can we use finite state automata to evaluate l's complement of a binary number? Design a machine to perform the same. (3Marks)(CS(R), Dec 2017)	**K3**
40.	Define Two-way finite automata. (3Marks)(CS(R), Dec 2017)	**K1**
41.	a) Construct Regular grammar for the regular expression L = (a + b)*(aa bb)(a + b)* (5Marks) b) List the closure properties of Regular sets. (4Marks)(CS(R), Dec 2017)	**K3**
42.	State Myhill-Nerode theorem. Minimize the following DFA by table filling method using Myhill-Nerode theorem describing the steps in detail. (9Marks)(CS(R), Dec 2017) 	**K3**
43.	Define regular grammar with suitable example. (3Marks)(CS(S), April 2018)	**K1**
44.	Prove the equivalence of non-deterministic finite automata with ε moves and regular expressions. (9Marks)(CS(S), April 2018)	**K2**
45.	a) Construct non deterministic finite automata (with ε moves) for regular (expression $(0+1)^*1$. (4Marks) b) Compare and contrast Moore and Mealy machines. (Justify with diagrams). (5Marks)(CS(S), April 2018)	**K3**
46.	a) Define the following (i) Language (ii) Finite Automata (iii) Transition diagram (3Marks)	**K3**

<table>
<tr><td></td><td>b) Design a moore machine which count the number of occurrences of substring aab in a given string (7Marks)

c) Prove that L is accepted by an NFA if and only if L is accepted by DFA (5Marks)(IT(R), 2017)</td><td></td></tr>
<tr><td>47.</td><td>a) Discuss about Chomsky classification of language (5Marks)

b) Prove the equivalence of moore and mealy machine (5Marks)

c) Construct a DFA that accepts all strings on {0,1}, except those containing the substring 001. (5marks)(IT(R), 2017)</td><td>K3</td></tr>
<tr><td>48.</td><td>a) Minimize the following DFA (7Marks)

<table>
<tr><th></th><th>0</th><th>1</th></tr>
<tr><td>A</td><td>B</td><td>C</td></tr>
<tr><td>B</td><td>D</td><td>E</td></tr>
<tr><td>C</td><td>F</td><td>G</td></tr>
<tr><td>*D</td><td>D</td><td>E</td></tr>
<tr><td>E</td><td>F</td><td>G</td></tr>
<tr><td>*F</td><td>D</td><td>E</td></tr>
<tr><td>*G</td><td>F</td><td>G</td></tr>
</table>

b) Prove that L is accepted by an NFA-sif and only if L is accepted by NFA (5Marks)

c) Define the following
(i) Kleene star
(ii) Concatenation
(iii Reversal (3Marks)(IT(R), 2017)</td><td>K2</td></tr>
<tr><td>49.</td><td>a) Describe the language of the following DFA. (4Marks)

<table>
<tr><th>δ</th><th>0</th><th>1</th></tr>
<tr><td>A</td><td>B</td><td>A</td></tr>
<tr><td>*B</td><td>A</td><td>B</td></tr>
</table>

b) State and prove the equivalence of NFA and DFA. (6Marks)

c) Design a Mealy machine to print 2’s complement of a binary number. (5Marks)(IT(S), 2017)</td><td>K3</td></tr>
</table>

MODULE 3

Pumping Lemma for Regular Languages, Applications of Pumping Lemma. Closure Properties of Regular sets (Proofs not required), Decision Problems related with Type 3 Formalism Type 2 Formalism: - Context-Free Languages (CFL), Context- Free Grammar (CFG), Derivation trees, Ambiguity, Simplification of CFG, Chomsky Normal Form, Greibach normal forms. (CO1)		
Sl No.	**Questions**	**Knowledge Level**
1.	Prove that L = { 0^n/ n is a perfect square } is not regular by applying Pumping Lemma for Regular Languages.	**K2**
2.	Prove using pumping lemma that the language L= {a^nba^n ; n>=0} is not regular.	**K2**
3.	Prove using pumping lemma that the language L= {$0^n1^m2^n$; m,n>=0} is not regular	**K2**
4.	What is an ambiguous CFG? Illustrate with an example.	**K1**
5.	Write a grammar for the L = { ww^R / w in (0+1)*} language L and convert it into Chomsky Normal Form	**K1**
6.	What are the applications of context free languages ?	**K1**
7.	Prove that if 'W' is a string of a language then there is a parse tree with yield W' and also prove that if A = W then it implies that TV'is a sting of the language L defined by a CFG.	**K2**
8.	Construction of reduced grammar.	**K3**
9.	Elimination of null productions' In a CFG productions of the form A + can be eliminated where A is a variable.	**K2**
10.	Elimination of unit productions' In a CFG productions of the form A + B can be eliminated, where A and B are variables'	**K2**
11.	Elimination of Useless Symbols' These are the variables in CGF which does not derive any terminal or not reachable form the start symbols. These can also eliminated'	**K2**
12.	Explain in detail the ambiguity in context free grammar	**K2**
13.	State pumping lemma and its advantages.	**K2**
14.	Prove using pumping lemma that the given languages are not regular. a) L = {$a^{n!}$|n>0} b) L = {0^n1^{2n}|n>0} c) L = {ww^R|w Є(0,1)*} d) Balanced parenthesis. e) L = {0^n10^{2n}|n>=0}	**K2**
15.	Define Context-Free Grammar (CFG) with an example. [Dec 2007]	**K1**
16.	Construct a CFG for the given language L= {0^n1^n / n>=1}. [Nov 2011]	**K3**

17.	Construct a CFG for the given regular expression r =(011+1)*(01)*.	**K3**
18.	Define parse tree with an example?	**K1**
19.	What you meant by ambiguous grammar? Write an example.	**K1**
20.	What are useless symbols in CFG? [Dec 2007]	**K1**
21.	What are null and unit productions in CFG?	**K1**
22.	Define Chomsky Normal Form (CNF.) [May 2008]	**K1**
23.	Define Greibach Normal Form (GNF). [May 2008]	**K1**
24.	Construct a CFG for the given language L= $\{\{a^nb^n$ / n>=1} U $\{a^mb^{2m}$ / m>=1} }[May 2010]	**K3**
25.	Construct a CFG for the given regular expression r=(a+b)*aa(a+b)* and trace the derivation of the string ‘abaabb’.	**K3**
26.	Construct a CFG for the language of a‘s and b’s with exactly one ’a’.	**K3**
27.	Construct a CFG to generate set of even length strings with the two middle symbols same.	**K3**
28.	Show that this G ({S}, {a, b,+,*}, P,S) where P consists of S->S+S/S*S/a/b is ambiguous.	**K2**
29.	Explain the steps for CFG simplification.	**K2**
30.	Remove useless symbols from the given grammar. A->xyz/Xyzz X->Xz/xYz Y->yYy/Xz Z->Zy/z	**K2**
31.	Remove unit productions from the given grammar. S->AB A->a B->C/b C->D D->E E->a	**K2**
32.	Remove null productions from the given grammar. S->AB A->a/€ B->b/€	**K2**

33.	Convert the given CFG into CNF S->bA/aB A->bAA/aS/a B->aBB/bS/a	**K3**
34.	Convert the given CFG to GNF S->ABA A->aA/€ B->bB/€	**K3**
35.	Prove that the language $L=\{a^n b^n c^n, n>=0\}$ is not context free.	**K2**
36.	Explain the process of simplification and simplify the given CFG S->XYac X->YC Y->b/€ C->D/€ D->d	**K2**
37.	(a) Explain elimination of useless symbols form a given CFG. Eliminate useless symbols from the given grammar S->AB/a A->b (b) Explain the elimination of null productions. Eliminate all null productions from the given grammar. S->ABAC A->aA/€ B->bB/€ C->c	**K2**
38.	(a) Explain Chomsky Normal Form(CNF). Convert the given CFG into CNF S->1A/0B A->1AA/0S/0 B->0BB/1 (b) Explain Greibach Normal Form (GNF). Convert the given CFG to GNF. S->AB A->aA/bB/b B->b	**K2**

39.	Write CFG for generating the following language: (1) To generate palindromes for binary numbers. (2) Strings with equal number of a's and b's. (3) L = {w/ $n_a(w) = 2n_b(w)$} (4) Strings of a's and b's with at least one 'a'. (5) Set of strings with first, middle, and last symbols are same.	**K1**
40.	Determine whether S->aSb/aA/€, A->b is ambiguous or not.	**K2**
41.	Eliminate useless symbols from the following CFG: S->aB/bX, A->BAd/bSX/a, B>aSB/bBX, X->SBD/aBx/ad S->aC//SB, A>bSCa, B>aSBbBC, C->aBC/ad	**K2**
42.	Eliminate unit productions from the following CFG: I->a/b/Ia/Ib/I0/I1, F->I/(E), T->F/T*F, E->T/E+T S->XY, X->a, Y->Z/b, Z->A, A->B, B->c	**K2**
43.	Eliminate null productions from the following CFG: S->aSa/bSb/€ E->AXA, A->aA/€, X->bX/€ S->ABAC, A->aA/€, B->bB/€, C->c	**K2**
44.	Reduce the following G in to GNF S->AB, A->BS/b, B->SA/a.	**K3**
45.	Convert the given CFG to GNF: S->BB/b, B->SS/b S->AaBb, A->abB/€, B->Abb/A/€	**K3**
46.	Convert the given CFG to GNF: S->BB/b, B->SS/b S->AaBb, A->abB/€, B->Abb/A/€	**K3**
47.	Which Normal Form representation of CFG will you prefer in converting CFG to NPDA? Why? (3Marks)(CS(R), Dec 2017)	**K2**
48.	What do you mean by useless symbol in a grammar? Show the elimination of useless symbols with an example. (3Marks)(CS(R), Dec 2017)	**K2**

49.	Define CFG for the following languages over the alphabets {a,b} (9Marks)(CS(R), Dec 2017) (i) L={ $a^{m+n}b^mc^n$ n,m>0} (ii) L contains all odd length strings only (iii) L = {$0^n1^n2^n$ n>0}	**K2**
50.	Construct context free grammar for L= {wcwR | w in (a+b)*}, Reverse of w is denoted as wR. (3Marks)(CS(S), April 2018)	**K3**
51.	List conditions for symbols to become *useful* symbols in context free grammar. (3Marks)(CS(S), April 2018)	**K1**
52.	Do the following: (9Marks)(CS(S), April 2018) i) Derive any two representative strings with minimum length 4 from following context free grammar. G= ({S,A,B},{a,b},P,S) S□ bA | aB A□bAA | aS | a B□aBB | bS | b ii) Draw derivation tree corresponding to string aabbab with respect to aforementioned grammar.	**K2**
53.	a) Use pumping lemma to show that an | n is a perfect cube is not a CFL (5Marks) b) Prove that regular expression is closed under homomorphism (5Marks) c) Give CFG for the following regular expression (0+11)*(011)1* (5Marks) (IT(R), April 2018)	**K2**
54.	a) Give regular expressions for the following: (2Marks) i) Set of all binary strings beginning with 110. ii) Set of all binary strings, contains exactly three 1's. b) Convert the following regular expression to ε-NFA and then to NFA. i) 011(0+1)*(0+1) ii) (a+b)(ab)* (10Marks) c) Define Context Free Grammar and Context Free Language. (3Marks) (IT(S), April 2018)	**K2**

MODULE 4

Non-Deterministic Pushdown Automata (NPDA), design. Equivalence of acceptance by final state and empty stack in PDA. Equivalence between NPDA and CFG, Deterministic Push Down Automata, Closure properties of CFLs (Proof not required), Decision Problems related with Type 3 Formalism. (CO1, CO3)		
Sl No.	**Questions**	**Knowledge Level**
1.	Design a PDA which accepts L = { ww^R / w in (0+1)*}	**K3**
2.	What is the acceptance concept of push down automata?	**K1**
3.	Define Push Down Automata (PDA). [Nov 2011]	**K1**
4.	Give instantaneous description of PDA.	**K1**
5.	Give an example of language accepted by a PDA but not by DPDA. [Nov 2009]	**K1**
6.	Design a PDA for the language {a^nb^{2n}/n>=0} [May 2013]	**K3**
7.	Explain the components of PDA with a neat diagram.	**K2**
8.	Compare PDA with finite automata. [Dec 2012]	**K2**
9.	Differentiate language acceptability by PDA through empty stack and final state. [May 2008]	**K2**
10.	What is PDA? Give formal definition of PDA and explain the components and working of PDA. Compare PDA and FA with an example.	**K2**
11.	For the grammar S->aABC, A->aB/, B->bA/b, C->a, obtain the corresponding PDA and trace for the string w=aabaa. [May 2013]	**K3**
12.	Construct PDA to accept the language given by {w € (a,b)*/ w has the same number of a's and b's }[Dec 2007].	**K3**
13.	Design a PDA for the language L={WCW^R / W ϵ (0+1)*} by empty stack. [May 2010]	**K3**
14.	Design a PDA that accept the language L={0^n1^n/n>=0} [Dec 2012]	**K3**
15.	Construct PDA for the language L= { x € (a,b)*/ $n_a(x) > n_b(x)$ } [May 2008]	**K3**
16.	Distinguish between deterministic and nondeterministic PDA with examples. [May 2010]	**K2**
17.	List some applications of PDA. [May 2008], [Dec 2012]	**K2**

18.	Design PDA for each of the following requirements: 1) L = $\{a^i b^j c^k$ / i=j+k; i, j, k>=0} 2) L = $\{0^n 1^m 0^n$ / m, n>0} 3) L = $\{0^n 1^{n+2}$ / n>=1} 4) L = {uawb: u & w € (a,b)* and |u| =|w|}	**K3**
19.	Show that a PDA M given below is deterministic. M = ({q0, q1}, {0, 1, 2}, δ, q0, Z0, {q1}), where δ is defined as δ(q0, 0, Z0) = (q0, 0Z0) δ(q0, 1, Zo) = (q0, 1Zo) δ(q0, 0, 0) = (q0, 00) δ(q0, 1, 0) = (q0, 10) δ(q0, 0, 1) = (q0, 01) δ(q0, 1, 1) = (q0, 1Zo) δ(q0, 2, 0) = (q1, 0) δ(q0, 2, 1) = (q1, 1) δ(q1, 0, 0) = (q1, €) δ(q1, 1, 1) = (q1, €) δ(q1, €, Zo) =(q1, Z0) δ(q0, 2, Zo) = (q1, Zo)	**K2**
20.	Consider the PDA having no final state δ defined as: δ(q0, a, Zo) = (q0, aZo) δ(q0, a, a) = (q0, aa) δ(q0, b, a) = (q1, €) δ(q1, b, a) = (q1, €) δ(q1, €, Zo) = (q1, €) Show that the string $a^n b^n$ is accepted by PDA M by empty stack	**K2**
21.	Explain the different methods by which a PDA accepts a language. (3Marks)(CS(R), Dec 2017)	**K2**
22.	Can we construct a Deterministic PDA for the language ww^R? Justify your answer. Otherwise how can we modify this language to make it accepted by DPDA. (3Marks)(CS(R), Dec 2017)	**K2**
23.	Design a Push Down Automata for the language L=$\{a^n b^{2n}$ | n>0}. Trace your PDA with n=3. (9Marks)(CS(R), Dec 2017)	**K3**
24.	Prove that the following languages are not regular (9Marks) (CS(R), Dec 2017) (i) L = {0^(i^2) such that i ≥ 1} is not regular} (ii) L = $\{a^p$ such that p is a prime number}	**K2**
25.	List conditions required for push down automata to qualify as deterministic push down automata. (3Marks) (CS(S), April 2018)	**K1**
26.	List closure properties of context free language. (3Marks) (CS(S), April 2018)	**K1**

27.	Do the following: i) Construct push down automata with empty stack as final condition for Context free language, L= { wcw^R| w in (a+b)*} .Reverse of w is denoted as w^R. ii) Describe all instantaneous descriptions from initial ID (start state,abcba, initial stack symbol) |--* to final ID (state, ε, ε) with respect to constructed push down automata. (9Marks)(CS(S), April 2018)	**K3**
28.	Prove the equivalence of push down automata and context free grammar. (9Marks)(CS(S), April 2018)	**K2**
29.	a) What is a regular expression? Write a regular expression that accept all strings on {0,1} such that it accepts at most one pair of consecutive 1's (5Marks) b) What is ambiguous CFG ? Show that the grammar $E \to E + E \vert E * E \vert (E) \vert I$ $I \to a \vert b \vert c$ is ambiguous (5Marks) c) Design a PDA to accept L= {$0^n1^m0^m1^n$ |m,n≥ 1} (5Marks)(IT(R), Dec 2017)	**K3**
30.	a) State pumping lemma for regular languages. Use pumping lemma to show that L={a^p|p is a prime} is not regular (8Marks) b) Convert the grammar $S \to AB$, $A \to BS \vert b$, $B \to SA \vert a$ into GNF (4Marks) c) Construct the PDA equivalent to the following grammar $S \to 0BB$, $B \to 0S \vert 1S \vert 0$ (3Marks)(IT(R), Dec 2017)	**K3**
31.	a) Design Turing machine to compute addition of two numbers. Assume unary notation for number representation. (6Marks) b) Describe all instantaneous descriptions (ID) from initial ID: q0010 to Final ID: 00 with respect to constructed Turing Machine.(assume q0 as initial state.) (4Marks)(CS(S), April 2018)	**K3**
32.	a) Prove that for every regular expression, there exists a deterministic Finite Automata (8Marks). b) Show that the language L={0^n1^{2n}|n ≥ 1} is not regular. (7Marks)(IT(S), April 2018)	**K2**
33.	a) List the applications of PDA and CFL. (4Marks) b) Design a PDA for the language L={$a^ib^jc^k$|i≠j or j≠k}. (8Marks) c) Explain ambiguity in CFG with the help of an example. (3Marks)(IT(S), April 2018)	**K3**

MODULE 5

Pumping Lemma for CFLs, Applications of Pumping Lemma. Type 1 Formalism: Context-sensitive Grammar. Linear Bounded Automata (Design not required) Type 0 Formalism: Turing Machine (TM) – Basics and formal definition, TMs as language acceptors, TMs as Transducers, Designing Turing Machines. (CO1, CO4)		
Sl No.	**Questions**	**Knowledge Level**
1.	What is a turing machine?	**K1**
2.	State and prove the Pumping Lemma for Context-Free languages.	**K2**
3.	Prove that the language L={$a^n b^n c^n$, n>=0} is not context free.	**K2**
4.	Design a Turing Machine which accepts L = { a^n b^n | n > 0 }	**K3**
5.	Construct a turing machine to accept the language $a^n b^n c^n$.	**K3**
6.	Construct Turing machine to perform multiplication.	**K3**
7.	Design a Turing Machine which computes m-n, where m and n are integers.	**K3**
8.	Explain the instantaneous description of a Turing machine.	**K2**
9.	Design a Turing machine that computes a function f(m,n)=m+n in addition of two integers	**K3**
10.	What are the languages accepted by a Turing machine?	**K1**
11.	Design a Turing machine with the initial tape as 01110111110... and the output pattern 01111101110...	**K3**
12.	Design a Turing machine to compute a function f where, f :∑ ∑−⟶ ∑0= ∑1 = {a, b}, f (w) = w, w is the result of replacing an occurrence of a in wbyb and vice versa.	**K3**
13.	What is configuration of a Turing machine?	**K1**
14.	Define turing machine.	**K1**
15.	Design a Turing machine to compute n mod 2	**K3**
16.	Describe the action of a turing machine.	**K1**
17.	Construct a turing machine that increments a binary number.	**K3**
18.	Construct a turing machine that decrements a binary number.	**K3**
19.	Construct a turing machine that adds two unary numbers.	**K3**
20.	Construct a turing machine which computes the function f(n)=n+2 over unary numbers.	**K3**
21.	Construct a turing machine that decides the language L = {an bn |n ≥ 0}.	**K3**
22.	Construct a turing machine that shifts the input string one position to the left.	**K3**
23.	Construct a turing machine that accepts the language L = {w ∈ (a, b)∗ |w has equal number of a's and b's}.	**K3**
24.	Design a turing machine to recognise the language L = {0n 1n 0n |n ≥ 1}.	**K3**
25.	Construct a turing machine that will compute f(x,y)=x+y.	**K3**
26.	Design a turing machine M to recognise the language L = {wwR |w is in (a + b)∗}.	**K3**
27.	State and prove pumping lemma for Context Free Languages. (10Marks)(CS(R), Dec 2017)	**K2**
28.	Construct a Turing machine that recognzes the language L = { $a^n b^n c^n$ | n>0} (10Marks)(CS(R), Dec 2017)	**K3**
29.	a) What is a Context sensitive grammar (CSG). Design a CSG to accept the language L = {$0^n 1^n 2^n$ | n>0}(6Marks) b) Define Linear Bound Automata. (4Marks)(CS(R), Dec 2017)	**K3**

30.	a) State pumping Lemma for context free language (5Marks) b) Define formally Turing machine Model. (5Marks)(CS(S), April 2018)	**K2**
31.	a) Design Turing machine to accept language L={0^n1^n | n >=1} (6Marks) b) Describe all instantaneous descriptions (ID) from initial ID q001 to Final ID with respect to constructed TM. Assume q0 as start state. (4Marks)(CS(S), April 2018)	**K3**

MODULE 6

Variants of TMs -Universal Turing Machine, Multitape TMs, Non-Deterministic TMs, Enumeration Machine (Equivalence not required), Recursively Enumerable Languages, Recursive languages, Properties of Recursively Enumerable Languages and Recursive Languages, Decidability and Halting Problem. Chomsky Hierarchy. (CO1, C04, CO5)		
Sl No.	**Questions**	**Knowledge Level**
1.	What are the concepts used in UTMs ?	**K1**
2.	Prove that a language L is recognised by a Turing machine with a two-way infinite tape iff it is recognised by a turing machine with a one way infinite tape.	**K2**
3.	Write short notes on any two variants of turing machines	**K1**
4.	What is multi-head turing machine?	**K1**
5.	Construct a turing machine that accepts the language given by {wwR |w is in (0+1)}.	**K3**
6.	Explain universal turing machine and explain its applications	**K2**
7.	Explain briefly how to enumerate all possible turing machines computations, so that a given computation can be characterised by a single natural number code C.	**K2**
8.	Prove the equivalence of two-way infinite tape with standard turing machine	**K2**
9.	Design a m-tape Turing machine that works as a copying machine. [MGU April 2011]	**K3**
10.	What is a universal turing machine?	**K1**
11.	When do we say that a function is Turing computable?	**K2**
12.	Show how the language L = { ww^R/ w in (0+1)*} can be recognized using the features of a multi-tape Turing Machine.	**K2**
13.	What is meant by halting of a TM?	**K1**
14.	When is a problem said to be undecidable? Explain the Halting problem. Or Explain the halting problem of Turing machine. Prove that it is undecidable. [MGU Nov 2011]	**K2**
15.	When we say a problem is decidable? Give an example of undecidable problem.	**K2**
16.	Prove that if a language L and its complement are both recursively enumerable, then both L and its complement are recursive.	**K2**
17.	List the Chomsky classification of languages and grammars.	**K1**
18.	Explain Chomsky classification of languages.	**K2**
19.	What is “Universal Language”? Is it recursive? Why?	**K1**
20.	Explain Chomsky and Extended Chomsky Hierarchy,	**K2**

21.	Which of the following statements is/are FALSE? 1.For every non-deterministic TM, there exists an equivalent deterministic TM. 2.Turing recognizable languages are closed under union and complementation. 3.Turing decidable languages are closed under intersection and complementation. 4.Turing recognizable languages are closed under union and intersection. A.1 and 4 B.1 and 3 C.2 D.3.	**K2**
22.	Let L be a language and L' be its complement. Which one of the following is NOT a viable possibility? A.Neither L nor L' is RE. B.One of L and L' is RE but not recursive; the other is not RE. C.Both L and L' are RE but not recursive. D.Both L and L' are recursive.	**K2**
23.	Let L1 be a recursive language, and let L2 be a recursively enumerable but not a recursive language. Which one of the following is TRUE? A.L1′ is recursive and L2′ is recursively enumerable B.L1′ is recursive and L2′ is not recursively enumerable C.L1′ and L2′ are recursively enumerable D.L1′ is recursively enumerable and L2′ is recursive	**K2**
24.	a) Write a note on Recursive Enumerable Languages (5Marks) b) Discuss about Universal Turing Machines (5Marks)(CS(R), Dec 2017)	**K1**
25.	a) Explain Chomsky's Hierarchy of Languages (6Marks) b) Let L = {x/ x E (a + b + c)* and $\|X\|_a = \|X\|_b = \|X\|_c$ }. What class of language does L belong? Why? What modification will you suggest in the grammar to accept this language? (4Marks)(CS(R), Dec 2017)	**K2**
26.	Discuss the Undecidable Problems About Turing Machines(10Marks)(CS(R), Dec 2017)	**K2**
27.	a) Explain the significance of universal Turing machine. (5Marks) b) Compare and contrast recursive and recursively enumerable languages. (5Marks)(CS(S), April 2018)	**K2**
28.	a) Prove that union of two recursive languages is recursive. (5Marks) b) Explain the significance of halting problem. (5Marks)(CS(S), April 2018)	**K2**
29.	a) Explain general notations for productions of each formal language from Chomsky hierarchy. (5Marks) b) Prove that complement of a recursive language is recursive. (5Marks)(CS(S), April 2018)	**K2**
30.	a) Write note on variants of Turing Machine. Show that multi tape TM is equivalent to single tape TM. (9Marks) b) Design a TM which finds 2's complement of a given number. (8Marks) c) Prove that the complement of recursive language is recursive. (3Marks) (IT(R), Dec 2017)	**K3**
31.	a) What is Linear Bound Automata?(5Marks) b) Construct T.M which accepts the language $L=\{a^nb^nc^n\|n\geq 1\}$(8Marks) c) Prove that the halting problem is undecidable. (7Marks)(IT(R), Dec 2017)	**K3**
32.	a) Explain post correspondence problem. (5Marks) b) Prove that universal language is recursively enumerable. (8Marks) c) Construct T.M which reverse a string (7Marks)(IT(R), Dec 2017)	**K3**

33.	a) Show that the Universal Language is not recursive. (10Marks) b) Design a Turing Machine for L={ww | w ϵ{0,1}*}. (10Marks)(IT(S), April 2018)	**K3**
34.	a) List and explain the variants of Turing Machine, and show that they are equivalent to a single tape Turing Machine. (12Marks) b) Design a Turing Machine that performs integer addition. (8Marks)(IT(S), April 2018)	**K3**
35.	a) Define Halting Problem and show that it is undecidable. (5Marks) b) What is Linear Bounded Automata? (5Marks) c) Build a Turing Machine that accepts the language $L=\{a^nb^{2n}\}$. (10Marks)(IT(S), April 2018)	**K3**

COURSE PROJECT

(CO6)		
Sl No.	**Questions**	**Knowledge Level**
1.	Working of ATM	**K4**
2.	Working of FAN Regulator	**K4**
3.	Working of Digital Lock	**K4**
4.	Working of Quadrotor mission	**K4**
5.	Working of Chocolate Vending Machine	**K4**
6.	State of a Traditional PC	**K4**
7.	State of Metro Train	**K4**
8.	States of Ordinary Television	**K4**
9.	State of Student in Class	**K4**
10.	State of Coffee Vending Machine	**K4**