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.

Estd. 2000

Department of Computer Science and Engineering
Question Bank - Academic Year (2021-22)

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

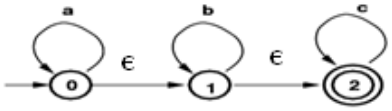
Name of the Faculty : M.Ganthimathi, ASP/cse, M.Azhagesan AP/cse

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

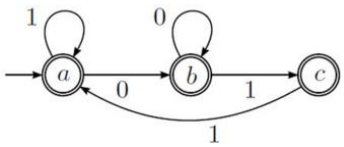
UNIT I - FINITE AUTOMATA

PART - A

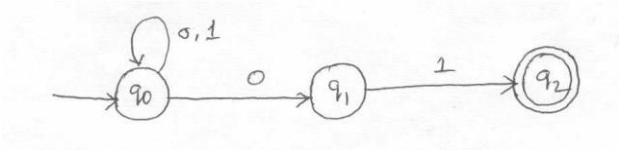
1. Define finite automata. What are the two types of FA .
2. Write the difference between the + closure and * closure
3. Define DFA.
4. Write the notations of NFA.
5. Write the Difference between DFA and NFA.
6. What are the applications of Finite automata?
7. Draw the DFA that accept over the alphabet 0,1 with 011 as a substring
8. Draw the NFA that accept over the alphabet 0,1 with 011 as a substring
9. Define Transition table & Transition diagram for the string end with 11 over ∑=(0,1)
10. Define NFA- ε.and Define ε- closure.
11. Compute the ε–closure of each state



12. Define the language of NFA.
13. Process the string for δ(a,0111) and $\bar{\delta}$ (a,1010)



14. Define extended transition function of DFA
15. Draw the transition diagram for an identifier
16. Design DFA to accept strings over ∑=(0,1) with two consecutive 0's.
17. Construct NFA-ϵ for 1*(01)*
18. Define M for the given automata

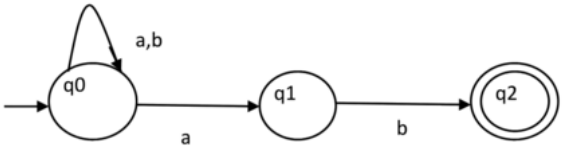


19. Differentiate NFA and NFA- ε. (CO1,K1)
20. Define M for the given automata

δ	a	b
→p	{p}	{p,q}
q	{r}	{r}
*r	-	-

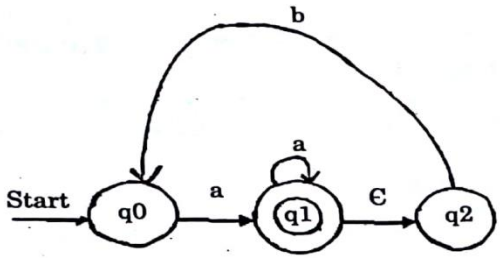
PART - B

1. (i) Define NFA and convert the following NFA to DFA .(8)



(ii) Prove that a language L is accepted by some DFA if L is accepted by some NFA.(8)

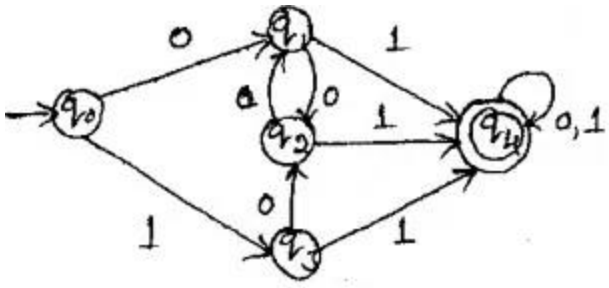
2. (i) Design DFA to accept language L={w/w has even number of 0's and even number 1's}.(8)
   (ii) Check whether the string 110101 , 10100 is accepted by the constructed DFA. (8)
3. (i) Construct NFA-ϵ that accepts all strings on {0,1} containing the substring 101.and convert it into NFA without ϵ. (8)
   (ii) If L is accepted by an NFA with ε-transition then show that L is accepted by an NFA without ε-transition.(8)
4. Construct NFA-ϵ for (a/b)*abb and convert it into DFA.(16)
5. Prove that a language L is accepted by some ε–NFA if and only if L is accepted by some DFA.(16)
6. Convert given NFA-ϵ into DFA..(16)



7. Construct a DFA equivalent to the following NFA.(16)

δ	a	b
→p	{p}	{p,q}
q	{r}	{r}
*r	-	-

8. Minimize the following DFA.(16)



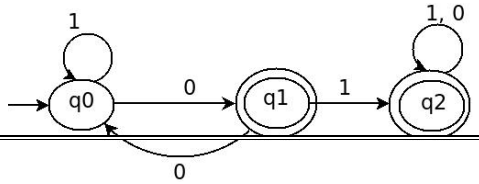
UNIT II - REGULAR LANGUAGES

PART - A

1. List the types of language
2. List out the recognizer of each language
3. Compare Finite automata and Push down automata
4. Write the Grammar rules for Regular language and Context free grammar
5. List the Chomsky hierarchy of languages
6. Write about type0 and type3 language
7. Write the rule for type 2 language
8. Define Regular expression
9. Define Regular language
10. Obtain a Regular expression to accept the strings of 0's and1's ending in 00
11. Obtain Regular expression such that L(r) ={w/w having no two consecutive 0 over ∑=(0,1)}
12. Construct NFA for RE R=(a/b)*abb
13. Define R_{ij} recursively
14. List out the applications of RE
15. State and Define pumping lemma for Regular Set
16. Prove L(r) ={w/w having equal number of 0 and 1 over ∑=(0,1)}is not regular
17. What are the closure properties of regular expression
18. Construct NFA for RE R=0*0+1
19. If L1=(a+b)*a and l2=b(a+b)*b then L1∩ L2 = ?
20. Let h(0)=bab , h(1)=ca*c. then h(010)=?

PART - B

1. Let R be Regular expression then there exists a Finite Automata M=(Q,∑, δ,q0,F) which accepts L®.(16)
2. Construct NFA For Regular Expression R=(01)*(10)*+00*.(16)
3. Let M=(Q,∑, δ,q0,F) be an FA then there exists an equivalent Regular Expression R for the language accepted by the finite automata.(16)
4. Convert given DFA to Regular Expression.(16)



5. Define pumping Lemma .Prove L(r) ={a^ib^j / i<=j }is not regular..(16)
6. Explain in detail about closure properties of regular sets.(16)
7. Illustrate the Chomsky grammar classification with necessary example..(16)

UNIT III - CONTEXT FREE GRAMMARS

PART - A

1. Define Grammar
2. List out the types of grammar
3. Define Context Free Grammars with example
4. Define derivation with example and list out types of derivation
5. Define Left most derivation and Right most derivation with example.
6. Define derivation tree or parse tree with example
7. Derive a string aababa for the following context free grammar S→aSX / b , X→Xb / a
8. What is L(G) ?.Find L(G) for G=({S},{0,1},{S->0S1,S->ε},S)
9. What is a ambiguous grammar? Give Example.
10. Show that E->E+E | E*E | (E) | id is ambiguous grammar for the string id+id*id
11. List out the steps to convert CFG to CNF.
12. What do you mean by null production and unit production? Give an example.
13. What are the two normal forms of CFG? Define CNF
14. What is meant by GNF give example.
15. Show that id+id*id can be generated by two distinct leftmost derivation in the grammar E->E+E | E*E | (E) | id .
16. Construct the CFG for generating the language L={a^n b^n /n>=1}.
17. What do you mean by Useless symbol?Give the steps to eliminate useless symbol with example
18. Find CFG with no useless symbols equivalent to : S→AB | CA ,B→BC | AB, A→a , C→aB | b.
19. Construct CFG without Є production from : S →a | Ab | aBa , A →b | Є , B →b | A.
20. Let G be the grammar S->aSbS / bSaS / Є obtain left most derivation tree for the string aababb

PART-B

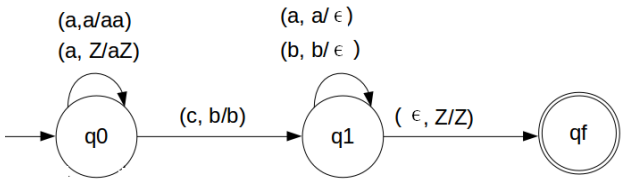
1. (i) Construct a CFG to generate even and odd set of palindromes over alphabet (a,b).(8)
   (ii) Generate CFG for the language L={$0^i1^j0^k$ / j>i+k}.(8)
2. (i) Obtain the CN F equivalent to the grammar S→bA/aB, A→bAA/aS/a, B→aBB/bS/b.(8)
   (ii) Eliminate unit production of the grammar S→A / bb, A→B /b, B→S/a.(8)
3. Construct a GNF grammar for the following S→AA / a, A→SS /b.(16)
4. Explain about parse tree.For the grammar S->aB/bA, A->a/aS/bAA, B->b/bS/aBB for the string aaabbabbba, (i) Find left most derivation. (ii) Right most derivation (iii) construct Parse tree (iv) Check ambiguity for the string abb.(16)

5.Let G=(V,T,P,S) be a CFG. Show that if S=α, then there is a derivation tree in a grammar G with yield α.(16)

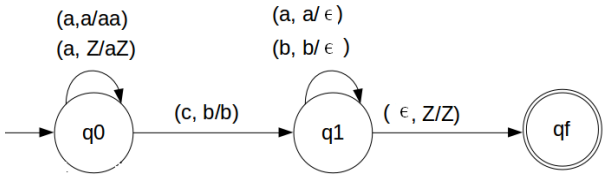
UNIT IV PUSHDOWN AUTOMATA

PART-A

1. Define the formal definition of PDA.
2. What is the language accepted by the Pushdown automata?
3. Define instantaneous description of Pushdown automata.
4. Define the Move of a PDA
5. Define the acceptance of a PDA by empty stack.
6. Write the steps to convert PDA to CFG.
7. Write the steps to convert CFG to PDA.
8. State pumping lemma for CFL.
9. State the closure properties of CFL
10. What is M=? for given PDA



11. Draw transition table for given PDA



12. What is the purpose of pumping lemma?
13. Show that L={a^p/p is prime }is not Context Free
14. Is it true that deterministic push down automata and non deterministic push down automata are equivalent in the sense of language of acceptances? Justify your answer.
15. What is additional feature PDA has when compared with NFA.
16. What are the different ways in which a PDA accepts the language?
17. Convert the following CFG to PDA S→AA , A →aS|bS|a.
18. Compare Deterministic and Non deterministic PDA.
19. Design a PDA for acceptation a language{ L=$a^n b^n$ |n>=1}.
20. What are the components of PDA?

21. What are the applications of CFG?

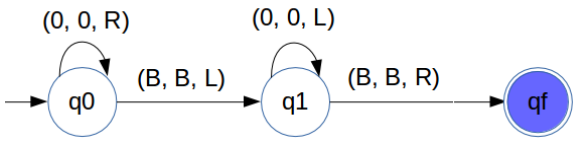
PART-B

1. Construct a PDA which accepts the language $L=\{a^{2n}b^n \mid n>=1\}$.(16)
2. Examine and construct a CFG G which accepts N(M), where
   M=({q0, q1},{a,b},{z0,z},d,q0,z0,F) and where dis given by
   δ(q0,b,z0)={(q0,zz0)}
   δ (q0, e,z0)={(q0, e)}
   δ (q0,b,z)={(q0,zz)}
   δ (q0,a,z)={(q1,z)}
   δ (q1,b,z)={(q1, e).(16)
3. Convert the grammar S ->0S1|A,A ->1A0|S|ε into PDA that accepts the same language by the Empty stack .Check whether 0101 belongs to N(M).(16)
4. Construct a PDA accepting $\{a^n b^m a^n / m, n>=1\}$ by empty stack. Also construct the corresponding Context-free grammar accepting the same set.(16)
5. State the Pumping Lemma for CFL and Develop the language $L=\{ a^nb^nc^n \mid n>=1\}$.(16)
6. If L is Context free language then prove that there exists PDA M such that L=N(M) .(16)
7. Construct PDA for the Language $a^nb^mc^{n+m}$.(16)
8. Explain in detail about Closure Properties of Context free Languages.(16)

UNIT V - TURING MACHINES

PART-A

1. Define formal definition of Turing machine
2. Define instantaneous description of a Turing Machine.
3. Give the configuration of TM
4. Define computable function
5. Draw the model of TM and write its feature
6. Define M and draw transition table for given TM



7. What is the role of checking off symbols in a Turing Machine?
8. Design a TM to implement the function f(x) = x+1. .

9. What are the techniques for TM construction?
10. Define multi head Turing machine.
11. What is the difference between Turing machine and finite control Turing machine?
12. Differentiate TM and PDA
13. What is Halting Problem.

14. What are the special features of TM?
15. Define multi tape Turing Machine.
16. Draw Turing machine for 010 as a substring over the alphabet 0 and 1.
17. Draw the Turing machine for f(x) =0, where x is an integer.
18. Define universal TM
19. Define Recursive and recursively enumerable languages
20. When is a recursive enumerable language said to be recursive
21. State when a problem is said to be decidable and give an example of an undecidable problem

PART-B

1. Design Turing machine for computing f(m,n)=m-n (proper subtraction).(16)
2. Design TM to accept the language L={$1^n 2^n$ | n >= 1 }.(16)
3. Design a Turing Machine M to implement the function "multiplication" using the subroutine 'copy'.(16)
4. i. Demonstrate the working of your TM with an example.(8)
   ii. Explain how the multiple tracks in a Turing Machine can be used for testing given positive integer is aeven or not..(8)
5. Explain halting problem .Is it solvable or unsolvable problem ?Discuss (16)
6. Prove that halting problem id decidable.(16)
7. Explain the properties of recursive Language.(16)
8. Show Lu is recursively enumerable but not recursive.(16)

Course Faculty **HoD**