|| Jai Sri Gurudev||

Sri Adichunchanagiri Shikshana Trust®

SJB INSTITUTE OF TECHNOLOGY

No. 67, BGS Health & Education City, Dr. Vishnuvardhan Road, Kengeri, Bengaluru -560060

Accredited by NAAC, Accredited by NBA. Certified by ISO 9001-2015

Department of Computer Science & Engineering

Question Bank

Subject: Automata Theory & Computability

Subject Code: 18CS54

Sem: V

VTU QUESTION BANK

Unit 1

Introduction to Finite Automata

. 1. Obtain DFAs to accept strings of a's and b's having exactly one a**.(5m)(Dec-2014)**
2. Obtain a DFA to accept strings of a's and b's having even number of a's and b's **.(5m)(Dec-2013)**
3. Give Applications of Finite Automata**. (5m)(June-July-2014)**
4. Define DFA, NFA & Language? **(5m)(Dec-2014)**
5. Obtain a DFA to accept strings of a's and b's starting with the string ab**. (6m)(Dec-2013) (June-July-2015)**
6. Draw a DFA to accept string of 0's and 1's ending with the string 011.**(4m)(Dec-2013) (June-July 2014)**
7. DFA to accept strings of 0's, 1's & 2's beginning with a 0 followed by odd number of 1's and ending with a 2. **(10m)(Dec-2013)(June-July-2015)**
8. Design a DFA to accept string of 0's & 1's when interpreted as binary numbers would be multiple of 3 **(5m)(June-July-2013, June-July-2014)** .
9. Find closure of each state and give the set of all strings of length 3 or less accepted by automaton.**(5m)(June-July-2013)**
10. Obtain a DFA to accept strings of a's and b's having a sub string aa. **(5m)(June-July-2013)**
11. Write Regular expression for the following L = { $a^n b^m$: m, n are even} L = { a^n,b^m : m>=2, **(5m)(Dec-2014, June-July-2014)**

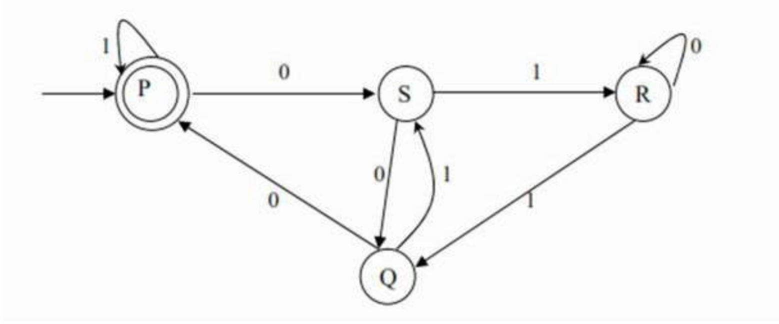
12. Convert above automaton to a NFA to DFA .(Dec-2013, June-2015)

δ		a	b
□ p	{r}	{q}	{p,r}
q	**I**	{p}	**I**
*r	{p,q}	{r}	{p}

13. Convert following NFA to DFA using subset construction method**.(10m)(June-July-2014**)

δ_N	0	1
□ p	{p,r}	{q}
q	{r,s}	{p}
*r	{p,s}	{r}
*s	{q,r}	**I**

14. Convert the following DFA to Regular Expression **(10m)(Dec -2014)**

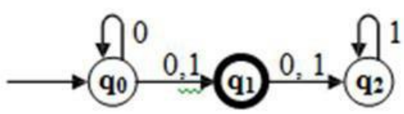


15. Define NFA. With example explain the extended transition function**(5m)(Dec-2014)**
16. Explain the ground rules of finite automata.**(5m)(June-july-2013)**

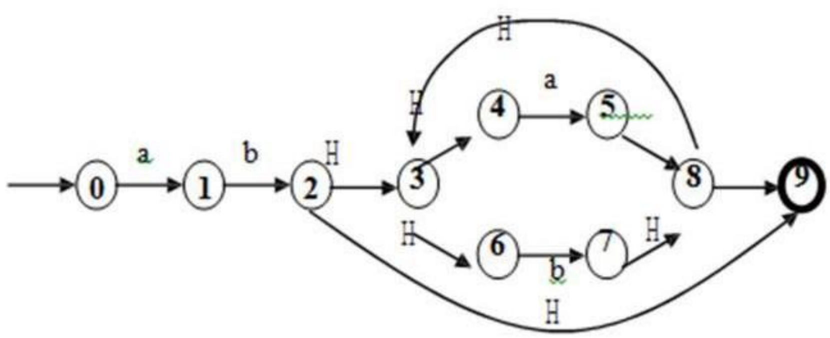
UNIT 2

Finite Automata, Regular Expressions

1. P.T. Let R be a regular expression. Then there exists a finite automaton M = (Q,¦, G, q_0, A) which accepts L(R). **(10m)(June- July2014)**
2. Define derivation , types of derivation , Derivation tree & ambiguous grammar. Give example for each. **(4m)(June- July 2015)**
3. Obtain an NFA to accept the following language L = {w | w abab^n or aba^n where n t 0} **(6m)(Dec-2013, June- July- 2014)**
4. Convert the following NFA into an equivalent DFA. **(10m)(June-July-2015)**



5. Convert the following NFA to its equivalent DFA**(10m) (June-July-2015)**



6. Obtain an NFA which accepts strings of a's and b's starting with the string ab. **(7m)(June- July 2013)**
7. Define grammar? Explain Chomsky Hierarchy? Give an example **(6m)(June- July 2014)**
8. Is the following grammar ambiguous **(7m)(Dec-2013, June-July-2014)**
   S -> aB | bA
   A -> aS | bAA |a

B -> bS | aBB | b

9. Obtain grammar to generate string consisting of any number of a's and b's with at least one b. **(5m)(Dec-2014, June-July-2015)**
10. Obtain a grammar to generate the following language L ={WWR Where W {a, b}*}. **(5m)(June-july-2015)**
11. Obtain a grammar to generate the following language: L = { $0^m 1^m 2^n$ | m>= 1 and n>=0}. **(5m)(June-July 2013)**
12. Obtain a grammar to generate the following language: **(5m)(Dec-2013)**

    L = { w | n $a^{(w)}$ > n $b^{(w)}$ }

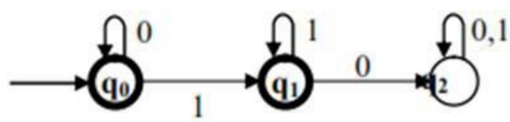
    L = { $a^n b^m c^k$ | n+2m = k for n>=0, m>=0}
13. Define PDA. Obtain PDA to accept the language L = {$a^n b^n$ | n>=1} by a final state. **(5m)(June-july-2014)**
14. Write a short note on application of context free grammar. **(7m)(Dec- 2014)**
15. Explain finite automata with epsilon transition. **(7m)(June-July- 2014)**
16. Explain the application of regular expression **(6m)(June-july- 2015)**

Unit 3

Regular Languages, Properties of Regular Languages

1. Prove pumping lemma? **5m)(June-July-2015, July-2014)**
2. Prove that L={w|w is a palindrome on {a,b}*} is not regular. i.e., L={aabaa, aba, abbbba,…} **(8m)(Dec-2014, June-July 2013)**
3. Prove that L={ all strings of 1's whose length is prime} is not regular. i.e., L={1^2,1^3 ,1^5 ,1^7 ,1^{11} ,----} **(8m)(Dec-2013)**

4. Let M = (Q, ¦, G, q0, A) be an FA recognizing the language L. Then there exists an equivalent regular expression R for the regular language L such that L = L(R). **(8m) (June-July-2015, Dec-2014)**
5. What is the language accepted by the following FA. **(6m)(June-July-2013)**



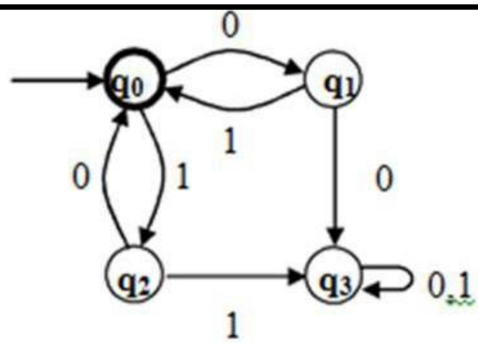
6. Show that following languages are not regular **(10m)(June-July 2014)**
   L={a^nb^m | n, m t0 and n<m }
   L={a^nb^m | n, m t0 and n>m }
   L={$a^nb^mc^md^n$ | n, m t1 }
   L={a^n | n is a perfect square }L={a^n | n is a perfect cube }

7. Apply pumping lemma to following languages and understand why we cannot complete proof **(10m)(Dec-2014)**

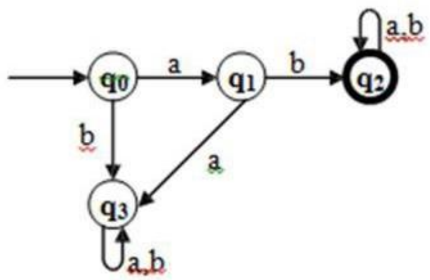
   L={anaba| n t0 }

   L={anbm | n, mt0 }

8. Obtain a DFA to accept strings of a's and b's starting with the string ab **(10m)(Dec-2013)**
9. Obtain a regular expression for the FA shown below**: (10m) (June-July-2015)**



10. Solve: **(10m)(Dec-14)**



11. **Explain Closure properties with an example. (10m)(June-July 2013)**

UNIT 4

Context-Free Grammars And Languages

1. P.T. If L and M are regular languages, then so is L⊂ M. **(10m)(June-July 2014)**
2. Write a DFA to accept the intersection of L1=(a+b)*a and L2=(a+b)*b that is for L1 ˆL2. **(10m)(June-July 2015, June-July-2013)**
3. Find the DFA to accept the intersection of L1=(a+b)*ab (a+b)* and L2=(a+b)*ba (a+b)* and that is for L1 ˆ L2 **(10m)(Dec-2013, June-July-2014)**
4. P.T. If L and M are regular languages, then so is L – M. **(10m)(Dec-2014)**
5. Design context-free grammar for the following cases**(10m)(Dec-2013, June-July 2014)**
   L={ 0n1n | n
   L={aibjck| i≠j or j≠k}
6. Generate grammar for RE 0*1(0+1)* **(10m)(June-July 2015)**
7. P.T. If L is a regular language over alphabet S, then L = 6* - L is also a regular language. **(8m)(June-July-2013)**
8. P.T. - If L is a regular language over alphabet 6, then, L = 6* - L is also a regular language. **(8m)(Dec-2013)**
9. P.T. If L is a regular language, so is L^R **(6m)(Dec-2014)**
10. . If L is a regular language over alphabet 6, and h is a homomorphism on 6, then h (L) is also regular. **(10m)(June-july- 2015).**
11. Explain CGF with an example. . **(5m)(June-July-2014)**
12. Explain decision properties of regular language. . **(5m)(June-July-2013)**

UNIT 5
Pushdown Automata

1. Give leftmost and rightmost derivations of the following strings
   a. 00101 b) 1001 c) 00011**(4m) (June-July 2015) (Dec-2013)**
2. Construct PDA: For the language **(4m)(June-July2014)**

   $L = \{wcw^R \mid w \quad \{a, b\}^*, c \quad \Sigma \quad \}$

3. Construct DPDA which accepts the language $L = \{wcw^R \mid w \quad \{a, b\}^*, c \quad \Sigma\}$. **(4m)(June-July 2013)**
4. Construct DPDA for the following: Accepting the language of balanced parentheses. (Consider any type of parentheses)Accepting strings with number of a's is more than number of b's Accepting {0n1m| n t m}**(June-July-2015)**
5. Design nPDA to accept the language: **(Dec-2013)**
   $\{a^i b^j c^k \mid i, j, k$ **t** 0 and $i = j$ or $i = k\}$
   $\{a^i b^j c^{i+j} \mid i, j$ **t** $0\}$
   $\{a^i b^{i+j} c^j \mid i$ **t** $0, j$ **t** $1\}$
6. Construct PDA: For the language **L={a^n b^{2n}|a,b €n t 0} (5m)(Dec-2015,July-2013)**
7. Construct PDA to accept if-else of a C program and convert it to CFG. (This does not accept if –if –else-else statements) **(5m)(Dec-2013)**
8. Show that deviation for the string aab is ambiguous. **(5m)(June-July 2014)**
9. Suppose h is the homomorphism from the alphabet {0,1,2} to the alphabet { a,b} defined by h(0) = a; h(1) = ab & h(2) = ba
   a) What is h(0120) ?
   b) What is h(21120) ?
   c) If L is the language L(01*2), what is h(L) ?
   d) If L is the language L(0+12), what is h(L) ?
   If L is the language L(a(ba)*) , what is h-1(L) ? **(5m)(Dec-2013)**

10. Design a PDA to accept the set of all strings of 0's and 1's such that no prefix has more 1's than 0's. **(5m)(June-July2015)**
11. Construct PDA: Accepting the set of all strings over {a, b} with equal number of a's and b's. Show the moves for abbaba**. (5m))(June-July 2013)**
12. Construct PDA: Accepting the language of balanced parentheses, (consider any type of parentheses). **(5m)(Dec-2014)**

13. How do you convert From PDA to CFG. **(5m)(Dec-2013)**

14. Convert PDA to CFG. PDA is given by P = ({p,q}, {0,1}, {X,Z}, δ, q, Z)), Transition function δ is defined by **(5m)(Dec-2014)**

δ(q, 1, Z) = {(q, XZ)}
δ(q, 1, X) = {(q, XX)}
δ(q, H, X) = {(q, H)}
δ(q, 0, X) = {(p, X)}
δ(p, 1, X) = {(p, H)}

15. Convert to PDA, CFG with productions **(10m)(Dec-2014)**

A o aAA, A -> aS | bS | a
S -> SS | (S) | H
S -> aAS | bAB | aB,
A -> bBB | aS | a,
B -> bA | a

16. Explain push down automata with an example**(10m)(Dec- 2013)**

UNIT 6

Properties of Context-Free Languages

1. Eliminate the n->n-generating symb->ls fr->m S -> aS | A | C, A ->a, B -> aa, C -> aCb. **(8m)(June-July 2015)**

2. Draw the dependency graph as given above. A is non-reachable from S. After eliminating A, G1= ({S}, {a}, {S -> a}, S). **(6m)(June-July 2013)**

3. Find out the grammar without H – Productions G = ({S, A, B, D}, {a}, {S o aS | AB, A -> H, B-> H, D ->b}, S). **(6m)(June-July 2014)**

4. Eliminate n->n-reachable symbols from G= ({S, A}, {a}, {S -> a, A ->a}, S) **(10m)(Dec-2013)**

5. Eliminate non-reachable symbols from S -> aS | A, A -> a, B -> aa. **(10m) (Dec-2014)**

6. Eliminate useless symbols from the grammar with productions S -> AB | CA, B ->BC | AB, A ->a, C -> AB | b. **(5m)(June-July 2014)**

7. Eliminate useless symbols from the grammar **(5m)(June-July 2015)**
P= {S o aAa, A ->Sb | bCC, C ->abb, E -> aC}
P= {S -> aBa | BC, A -> aC | BCC, C ->a, B -> bcc, D -> E, E ->d}
P= {S -> aAa, A -> bBB, B -> ab, C -> aB}
P= {S -> aS | AB, A -> bA, B -> AA}.

8. Write Algorithm to find nullable variables. **(5m)(June-July 2015)**

9. Eliminate H - productions from the grammar**. (5m)(Dec-2014, Dec-2013)**
S -> a |Xb | aYa, X -> Y| H, Y -> b | X
S -> Xa, X -> aX | bX | H
S -> XY, X ->Zb, Y -> bW, Z ->AB, W ->Z, A -> aA | bB | H, B -> Ba | Bb| H
S -> ASB | H, A -> aAS | a, B -> SbS | A| bb

10. Eliminate H - pr->ductions and useless symbols from the grammar S ->a |aA|B|C, A ->aB| H, B ->aA, C ->aCD, D ->dd. **(10m)(Dec-2014)**

11. Show that L = {$a^i b^i c^i$ | i **t**1} is not CFL. **(10m)(Dec-2013)**

12. Show that L = {ww |w {0, 1}*} is not CFL**. (10m)(June-july-2014)**

13. Using pumping lemma for CFL prove that below languages are not context free {p | p is a prime}. . **(10m)(Dec-2013)**

UNIT 7

Introduction To Turing Machine

1. Explain with example problems that Computers cannot solve.**(6m)(June-July-2015)**
2. Explain briefly the following Halting problem. **(4m)(June-July 2013)**
3. Explain Programming techniques for Turning Machines**(10m)(Dec-Jan-2010)**
4. Design a Turing machine to accept a Palindrome. **(10m)(Dec-2013)**
5. Design a TM to recognize a string of the form $a^n b^{2n.}$ **(10m)(Dec-2013)**
6. Design a Turing machine to accept a Palindrome. **(10m)(Dec--2014)**
7. Define undesirability, decidability. **(10m)(June-July 2014)**
8. Post's Correspondence problem Design a TM to recognize a string of 0s and 1s such that the number of 0s is not twice as that of 1s. **(10m) (June-July-2015, Dec-2013)**

UNIT 8

Undecidability

1. Design a TM to recognize a string of the form $a^n b^{2n.}$ **(10m) (June-July -2013)**
2. P.t If L is a recursive language, L is also recursive**.(10m) (June-July 2014)**
3. Design a Turing Machine to recognize 0n1n2n**. (10m)(Dec-2013)**
4. Explain briefly the following Halting problem**(6m)(Dec-2014, Dec-2013)**
5. Define undesirability, decidability. **(8m) (June-July 2013, June-July 2014)**
6. Post's Correspondence problem Design a TM to recognize a string of 0s and 1s such that the number of 0s is not twice as that of 1s. **(Dec-2014)**
7. Design a Turing machine to accept a Palindrome. **(7m)(Dec-2013)**
8. Write a short note on: **(20m) Dec-2014)**

a. Undesirability,
b. Halting problem,
c. decidability