QUESTION BANK

MODULE – I

1. Design a DFA which accepts strings with odd number of *0*'s and any number of *1*'s
2. Design a DFA which accepts strings with odd number of 0's and even number of 1's
3. Design a DFA which accepts strings with odd number of 0's and odd number of 1's
4. Design a DFA which accepts strings with even number of 0's and odd number of 1's
5. Design a DFA which accepts strings with even number of 0's and even number of 1's
6. Design a DFA which accepts strings with exactly two 0's
7. Design a DFA which accepts strings with abab as a sub-string
8. Design a DFA which accepts strings starts with 0 and ends with 1
9. Design a DFA which accepts strings with 00 as a sub-string but not 000 as a substring
10. Design a DFA which accepts strings with number of 0's is a multiple of 3
11. Design a DFA which accepts strings with second symbol is a and fourth symbol is b
12. Design a DFA which accepts strings with 101 or 110 as a sub-string
13. Design a DFA which accepts strings ending in 101 or 110
14. Design a DFA which accepts strings with every 00 (if exists) is followed by a 1
15. Design a DFA which accepts strings in which the left most symbol is differ from right most symbol.

MODULE – II

Each of the following languages is the intersection of two simpler languages. construct DFAs for the simpler languages, then combine them.

a. $\{w \mid w$ has at least three a's and at least two b's$\}$

b. $\{w \mid w$ has at exactly two a's and at least two b's$\}$

c. $\{w \mid w$ has an even number of a's and one or two b's$\}$

d. $\{w \mid w$ has an even number of a's and each a is followed by at least one b$\}$

e. $\{w \mid w$ has an even number of a's and one or two b's$\}$

f. $\{w \mid w$ has an odd number of a's and ends with a b$\}$

g. $\{w \mid w$ has even length and an odd number of a's$\}$

Each of the following languages is the complement of a simpler language. In each part, construct a DFA for the simpler language, then use it to give the state diagram of a DFA for the language given. In all parts $\Sigma = \{a, b\}$.

a. $\{w \mid w$ does not contain the substring $ab\}$
b. $\{w \mid w$ does not contain the substring $baba\}$
c. $\{w \mid w$ contains neither the substrings ab nor $ba\}$
d. $\{w \mid w$ is any string not in $a^*b^*\}$
e. $\{w \mid w$ is any string not in $(ab^+)^*\}$
f. $\{w \mid w$ is any string not in $a^* \cup b^*\}$
g. $\{w \mid w$ is any string that doesn't contain exactly two a's$\}$
h. $\{w \mid w$ is any string except a and $b\}$

Give state diagrams of NFAs with the specified number of states recognizing each of the following languages. In all parts the alphabet is $\{0,1\}$.

a. The language $\{w \mid w$ ends with $00\}$ with three states
b. The language of Exercise 1.6c with five states
c. The language of Exercise 1.6l with six states
d. The language $\{0\}$ with two states
e. The language $0^*1^*0^+$ with three states
f. The language $1^*(001^+)^*$ with three states
g. The language $\{\epsilon\}$ with one state
h. The language 0^* with one state

NFA to DFA Convertion

For a given N.F.A., construct D.F.A.

$$M = \{p, q, r, s\}, \{0, 1\}, \delta, p, \{s\}$$

Q \ Σ	0	1
p	p,q	p
q	r	r
r	s	–
s	s	s

Find D.F.A. accept equivalent to

$$M = (\{q_0, q_1, q_2\}, \{a, b\}, \delta, q_0 \{q_2\})$$

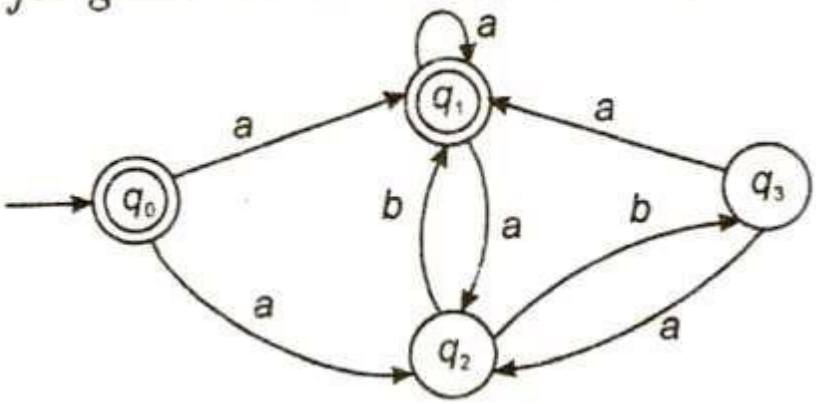
Q / Σ	a	b
q_0	q_0, q_1	q_2
q_1	q_0	q_1
q_2	–	q_0, q_1

Design D.F.A. for given N.F.A. where,

$$M = (\{p, q, r, s\}, \{0, 1\}, \delta, p, \{q, s\})$$

Q / Σ	0	1
p	q, s	q
q	r	q, r
r	s	p
s	–	p

Construct a D.F.A. for given N.F.A. transition Diagram.



MODULE – III

1. Write the Regular Expressions for the strings with exactly one a
2. Write the Regular Expressions for the strings with atleast two a's
3. Write the Regular Expressions for the strings begins with a or c followed by some number of b's
4. Write the Regular Expressions for any string in a or any string in c followed by any string in b
5. Write the Regular Expressions for the strings which should not contain double characters aa and bb
6. Write the Regular Expressions for the strings whose fifth symbol from right end is a
Write the Regular Expressions for the strings without bab as sub-string
8. Write the Regular Expressions for the strings with even number of 1's followed by odd number of 0's
9. Write the Regular Expressions for the strings begins with a and not having two consecutive b's
10. Write the Regular Expressions for the strings ending with either aa or b
11. Write the Regular Expressions for the strings with even number of a's followed by odd number of 1's
12. Write the Regular Expressions for the strings with 0's only or a single 0 followed by some 1's
13. Write the Regular Expressions for the language L={an bm | (m+n) is even}

MODULE – IV

1. Check whether the following languages are non-regular using pumping lemma for Regular Languages
   L={a2n | n>0}
   L={an2 | n>=0}
   L={0i1i | i>=1}
   L={anbncn | n>=1}
   L={ww | w ε (a,b)*}
2. Construct a CFG to generate balanced paranthesis
3. Construct a CFG to generate L={wcwR | w ε (a,b)*}
4. Construct a CFG to generate L={wwR | w ε (a,b)*}
5. Construct a CFG to generate a string containing a and b in any sequence
6. Construct a CFG to generate an alternating sequence of 0's and 1's

7. Construct a CFG to generate the string with no consecutive b's but only a's can be consecutive

MODULE – V

1. Check whether the following languages are non-context free using pumping lemma for Context Free Languages
   L={ap | p is prime}
   L={anbncn| n>0}
   L={aibj | j=i2}
   L={aibjck | i<j<k}
2. Design a PDA to accept the language "Odd Palindrome"
3. Design a PDA to accept the language "Even Palindrome"
4. Design a PDA to accept the language 0n12n+1
5. Design a PDA to accept the language L={0n1m | n>m>=0}
6. Design a PDA to accept the language L={anbman | n,m>=1}
7. Design a PDA to accept the language L={ambmcn | n,m>=1}
8. Design a PDA to accept the language L={ambn | n>m>=1}
9. Design a PDA to accept the language L={anbmcmdn | n,m>=1}