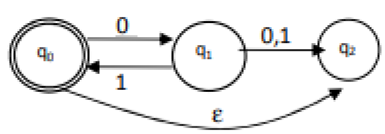
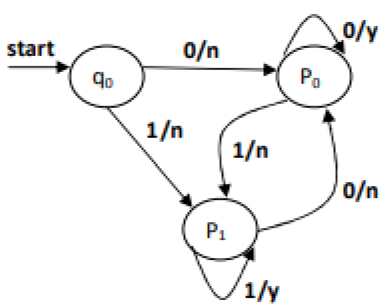
Unit I

1. Explain the design of a finite state machine with an example?
2. Explain the advantages of Finite State Machine?
3. What is NFA? Explain the transitions of NFA?
4. Design DFA to accept strings with ‘c’ and‘d’ such that number of d’s are divisible by 4.
5. Show with an example equivalence between NFA with and without €- transitions
6. Construct an NFA that accepts the set of all strings over {0,1} that start with 0 or 1 and end with 10 or 01.
7. Let $\Sigma = \{a, b\}$,
   a) Give DFA that accepts any string with aababb as a substring.
   b) Minimize the DFA obtained for the answer of question (a).
8. (0/1)*011 for this regular expression draw the NFA with ϵ-closures and convert it into NFA.
9. Construct a DFA equivalent to the NFA given below



10. Convert the following Mealy machine to an equivalent Moore machine



11. Differentiate between NFA and DFA?
12.

Construct Minimum state Automata for the following DFA?

* denotes final state

δ	0	1
➔ q1	q2	q6
q2	q1	q3
*q 3	q2	q4
q 4	q4	q2
q5	q4	q5
*q6	q5	q4

Unit II

1. Construct a DFA for the Regular Language consisting of any number of a's and b's
2. What is a regular language? Convert the given regular expression to regular language.

   i) $(1+\epsilon)(00^*1)0^*$
   ii) $(0^*1^*)000(0+1)^*$
   iii) $(00+10)^*1^*(10+00)^*$

3. What is relationship between finite automata and regular expression? Explain the process of converting DFA to regular expression.
4. Let Σ = {a, b}, a) Give DFA that accepts any string with aababb as a substring. b) Minimize the DFA obtained for the answer of question 2(a) using MyhillNerode theorem.
5. Construct DFA equivalent to regular expression (0+1)*(00+11)(0+1)* and also find the reduced DFA
6. Explain the Pumping Lemma for regular sets. Show that L={a^p | p is a prime} is not regular.

Unit III

1. Simplify the following CFG and Convert it into CNF
   S -> AaB | aaB
   A -> ε
   B -> bbA | ε
2. Explain different types of grammar with example?
3. Describe the closure properties of context free grammars. How to simplify the context free grammars? Explain
4. Obtain GNF equivalent to the grammar
   E->E+T/T,
   T->T*F/F,
   F->(E)/a ?
5. Define Context Free Grammar. State and Explain the closure properties of CFG
6. Show that language L={a^n b^ n c^n |n≥0} is not a Context Free.
7. Define Ambiguous Grammar? Check whether the grammar SaAB, AbC/cd, Ccd, Bc/d Is Ambiguous or not?
8. Generate left most and right most derivation and parse tree for given grammars
9. G1: S->0B|1A, A->0|0S|1AA, B->1|1S|0BB for the string 00110101 G2:S->Ab|bA, A->a|aS|bAA, B->b|bS|aBB for the string aaabbabbba
10. Find equivalent grammar in CNF for S->bA|aB, A->bAA|aS|a, B->aBB|bS|b
11. Define Ambiguous Grammar? Check whether the grammar S->aAB, A->bC/cd, C->cd, B->c/d Is Ambiguous or not?
12. Find GNF equivalent to the given CFG: E->E+T|T, T->T*F|F, F->(E)|id
13. Consider the CFG with {S,A,B} as the non-terminal alphabet, {a, b} as the terminal alphabet, S as the start symbol and the following set of production rules
    S → ASA | aB | b
    A → B

B → b | ∈ . Find a reduced grammar equivalent to the above grammar.

Unit IV

- Construct and explain a deterministic PDA for accepting language L= { 0^n 1^ n | n>=1}
- Explain about various components of PDA.
- What is Deterministic PDA? Differentiate acceptance by final state and acceptance by empty state
- Write an algorithm to obtain PDA from CFG with an example.
- Construct NPDA to accept all strings of a language L={ WW^R | W∈ (a+b)+ }
- List out the applications of Pushdown Automata
- How to convert the following grammar to PDA that accepts the same language by empty stack S ->0AA,A->0S/1S/0
- What is Deterministic PDA? Differentiate acceptance by final state and acceptance by empty state.
- What is deterministic Push Down Automata? Draw and explain a deterministic PDA for accepting { 0^n 1^ n | n>1}
- S->aABB|aAA, A->aBB|a, B->bBB|A, construct the PDA that accepts the language generated by given grammar.

Unit V

1. What are P and NP class of Languages? What is NP Complete and give examples?
2. Design a Turing Machine to accept the language L = $\{WW^R \mid W \in (a+b)^* \}$
3. Define Post Correspondence Problem? Explain in brief about PCP with an example?
4. What is decidability? Explain in brief about any two undecidable problems?
5. Design a Turing Machine to accept the language L = $\{a^n b^n c^n \mid n>0 \}$
6. Explain about Universal Turing Machine?
7. Discuss in brief about Turing reducibility?
8. Find whether post correspondence problem P={(10,101),(011,11),(101,011)} has match? Give the solution.
9. Write about Churches hypothesis and Computable function in Turing Machines with an example.
10. Explain the general structure of multi tape and non deterministic Turing machines and show that these are equivalent to basic Turing machines