**CS353-Theory of Computation**

**(A.Y. 2021-22)**

**Assignment**

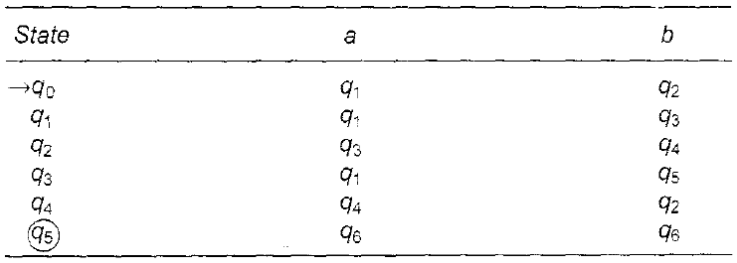
**Q. 1. Construct a Deterministic finite automaton equivalent to M=({p,q,r,s},{a,b}, d , q0,{s})**

|  |  |  |
| --- | --- | --- |
| State\ ∑ | a | b |
| p | p, q | p |
| q | r | r |
| r | s |  |
| s | s | s |

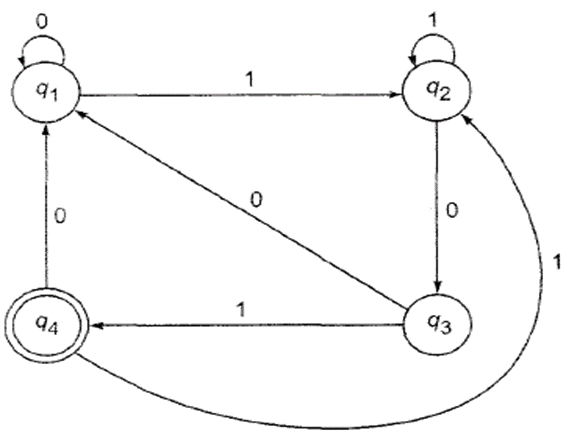
**Q. 2. Consider the Mealy machine describe by transition table given by following table. Construct a Moore machine which is equivalent to Mealy machine**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Present State | Next State | | | |
| a=0 | | a=1 | |
| State | Output | State | Output |
| q0 | q3 | 0 | q1 | 1 |
| q1 | q1 | 1 | q2 | 0 |
| q2 | q2 | 1 | q3 | 1 |
| q3 | q3 | 0 | q0 | 0 |

**Q. 3. Construct a minimum state automaton equivalent to the DFA describe by following fig.**



**Q. 4. Find the regular expression corresponding to following Fig.**



**Q. 5. Construct the finite automaton equivalent to the regular expression**

(0 + 1)\*(00 + 11)(0 + 1)\*

**Q. 6. Let G be the grammar S-> OB |1A, A -> 0 | 0S | 1AA, B -> 1|1S |OBB. For the string 00110101, find**

(a) the leftmost derivation,

(b) the rightmost derivation, and

(c) the derivation tree.

**Q. 7. If G is the grammar S -> SbS |a, show that G is ambiguous.**

**Q. 8. Let G be S -> AB, A -> a, B -> C I b, C -> D, D -> E and E -> a. Eliminate unit productions and get an equivalent grammar.**

**Q. 9. Reduce the following grammar G to Chomsky normal form (CNF). G is**

**S -> aAD, A -> aB | bAB, B -> b, D -> d.**

**Q. 10. Construct a grammar in Greibach normal form equivalent to the grammar**

**S ->AA | a, A -> SS | b.**

**Q. 11. Convert following Right Linear grammar to Left linear grammar**

**S -> aS | bA, A -> aS |bB, B -> aB | bB |** **ϵ**

**Q. 12. Eliminate Left recursion from following grammar**

**S -> ( L) | x, L -> L , S | S**

**Q. 13. Eliminate Left Factor from following grammar**

**S -> i E t S | i E t S e S | a, E -> b**

**Q. 14. Convert following grammar to finite automata**

**S -> 0A | 1A , A -> 0A | 1A |+B | -B, B -> 0B | 1B |0 |1**

**Q. 15. Construct PDA that accepts the language generated by following CFG**

**S -> SS | ( S ) | ( )**

**Q. 16. Construct PDA which recognizes following language**

**L = { WCWT  | W = { a, b}\* }**

**Q. 17. Design a Turing Machine for well-form parentheses.**

**Q. 18. Design a Turing Machine for odd length palindrome string.**