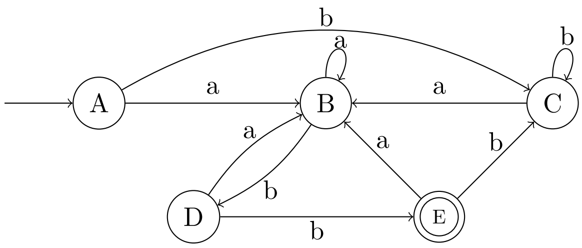
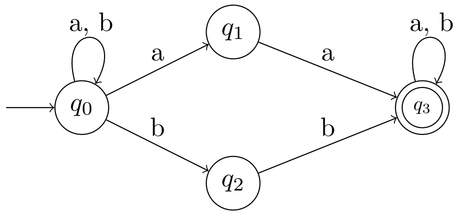
The equivalent DFA is



14. Construct a regular grammar for the RE, L = (a + b)*(aa + bb)(a + b)*.
    Solution: The NFA for the RE is



There are four states in the FA. So, in the regular grammar, there are four non-terminals. Let us take them as A (for q_0), B (for q_1), C (for q_2), and D (for q_3).
Now, we have to construct the production rules of the grammar.
For the state q_0, the production rules are

$$A \to aA,\ A \to bA,\ A \to aB,\ A \to bC.$$

For the state q_1, the production rules are

$$B \to aD,\ B \to a \text{ (as D is the final state).}$$

For the state q_2, the production rules are

$$C \to b\,D,\ C \to b \text{ (as D is the final state).}$$

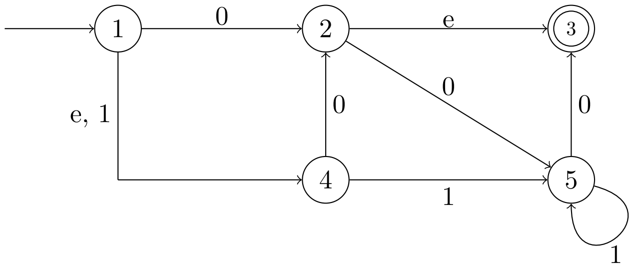
For the state q_3, the production rules are

$$D \to aD,\ D \to bD,\ D \to a/\ b.$$

The grammar = $\{V_N, \Sigma, P, S\}$

V_N = A, B, C, D Σ = a, b
P : A → aA/bA/aB/bC
B → aD/a
C → bD/b
D → aD/bD/a/b.

15. Consider the given FA and construct the smallest DFA which accepts the same language. Draw an RE and the grammar that generates it. [UPTU 2002]



Solution: This problem can be solved by the ∈ -closure method.

∈-closure(1) = {1, 4}   ∈-closure(3) = {3}   ∈-closure(5) = {5}

∈-closure(2) = {2, 3}   ∈-closure(4) = {4}

As 1 is the beginning state,start with ∈-closure(1) = 1,4.Let us rename it as A. A is still unmarked.

Then, construct a δ' function for the new unmarked state A for inputs 0 and 1.

δ'(A, 0) = ∈-closure(δ (A, 0))
= ∈-closure (δ ((1,4), 0))
= ∈-closure (2, 2)
= {2, 3}

It is a new state. Mark it as B.

δ'(A, 1) = ∈-closure(δ (A, 1))
= ∈-closure (δ ((1, 4), 1))
= ∈-closure (4, 5)
= ∈-closure (4) ∪ ∈-closure (5) = {4, 5}

It is a new state. Mark it as C.

δ'(B, 0) = ∈-closure(δ (B, 0))
= ∈-closure (δ ((2, 3), 0))
= ∈-closure (5)
= 5

It is a new state. Mark it as D.

δ'(B, 1) = ∈-closure(δ (B, 1))
= ∈-closure (δ ((2, 3), 1))
= ∈-closure ()
= ∅

δ'(C, 0) = ∈-closure(δ (C, 0))
= ∈-closure (δ ((4, 5), 0))
= ∈-closure (2, 3)
= B

δ'(C, 1) = ∈-closure(δ (C, 1))
= ∈-closure (δ ((4, 5), 1))
= ∈-closure (5)
= 5 = D

δ'(D, 0) = ∈-closure(δ (D, 0))
= ∈-closure (δ ((5), 0))
= ∈-closure (3)
= 3

It is a new state. Mark it as E.

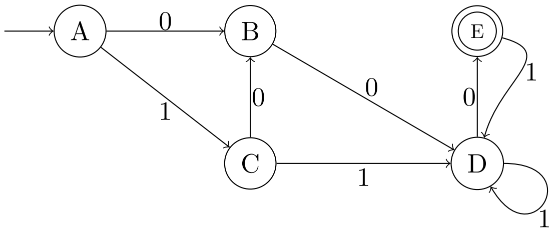
δ'(D, 1) = ∈-closure(δ (D, 1))
= ∈-closure (δ (5), 1)
= ∈-closure (5)
= 5 = D

δ'(E, 0) = ∈-closure(δ (E, 0))
= ∈-closure (δ (3), 0)
= ∅

δ'(E, 1) = ∈-closure(δ (E, 1))
= ∈-closure (δ (5), 1)
= ∈-closure (5)
= 5 = D

The beginning state is A, and the final state is E.

The transitional diagram of the DFA is



The RE is constructed using the Arden's theorem.

$$A = \wedge$$
$$B = 0A + 0C$$
$$C = 1A$$
$$D = 0B + 1C + 1D + 1E$$
$$E = 0D$$

Replacing A in B and C, we get

$$B = 0 + 0C$$
$$C = 1.$$

Replacing C in B, we get $B = 0 + 01$.
Replacing the new value of B, C, and E in D, we get

$$BD = 0(0 + 01) + 11 + 1D + 10D$$
$$B= 0(0 + 01) + 11 + D(1 + 10).$$

It is in the format of $R = Q + RP$, where $Q = (0 + (0 + 01) +11)$,$P=(1 + 10)$.
The solution is $R = QP^*$.

$$D = (0 + (0 + 01) + 11)\ (1 + 10)^*$$

Replacing the value of D in E, we get $E = 0(0 + (0 + 01) + 11)\ (1 + 10)^*$.
As E is the final state, the RE accepted by the FA is

$$0(0 + (0 + 01) + 11)\ (1 + 10)^*.$$

The regular grammar is constructed as follows taking each state into account. There are fi ve states in the FA. So, in the regular grammar, there are fi ve non-terminals.
Let us take them as A, B, C, D, and E according to the name of the states.
Now we have to construct the production rules of the grammar.
For the state A, the production rules are

$$A \to 0B,\ A \to 1C.$$

For the state B, the production rules are

$$B \to 0D$$

For the state C, the production rules are

$$C \to 0B \quad C \to 1D.$$

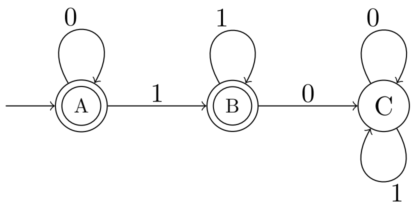
For the state D, the production rules are

$$D \to 1D \quad D \to 0E \quad D \to 0.$$

For the state E, the production rules are

$$E \to 1D.$$

16. Find the RE recognized by the finite state automaton of the following figure. [GATE 1994]



Solution: The equation for the FA is

$$A = 0A + \wedge \tag{1}$$
$$B = 1A + 1B \tag{2}$$
$$C = 0B + 0C + 1C \tag{3}$$

Solving the equation (1) using the Arden's theorem, we get $A = \wedge 0^* = 0^*$.
Putting the value of A in equation (2), we get

$$B = 10^* + 1B.$$

Using the Arden's theorem, we get

$$B = 10^*1^*.$$

Both A and B are final states, and thus the string accepted by the FA is

$$0^* + 10^*1^*$$
$$= 0^* (\wedge + 11^*) = 0^*1^* \text{ (as } \wedge + RR^* = R^*).$$

Multiple Choice Questions

1. The machine format of regular expression is:
a) Finite automata b) Push down automata c) Turing machine d) All of the above
2. The regular expression is accepted by:
a) Finite automata b) Push down automata c) Turing machine d) All of the above
3. The language of all words with at least 2 a's can be described by the regular expression:
a) (ab)*a b) (a + b)*ab*a(a + b)* c) b*ab*a(a + b)* d) All of the above
4. The set of all strings of {0, 1} having exactly two 0's is:
a) 1*01*01* b) {0 + 1)*1 c) {11 + 0}* d) {00 + 11}*