# MyHill Nerode Theorem

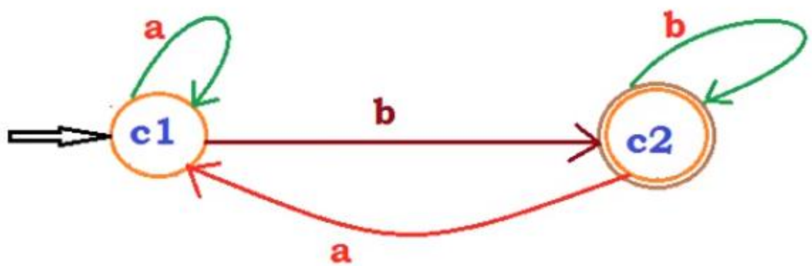
# MyHill Nerode Theorem

- A language L is regular if and only if the equivalence $R_L$ has a finite number of equivalence classes of strings and the number of states in the smallest DFA recognizing L is equal to the number of equivalence classes in $R_{L.}$

# Myhill Neorode Theorem

For a language L, defined over an alphabet∑, L partitions ∑* into distinct classes.
If L generates finite number of classes then L is regular.

**Question:**
**Let the language L of all strings , ending with b, defined over ∑ = {a,b}**




It can be observed that L partitions ∑* into the following classes:

C1 = set of all strings ending in a.
C2= set of all strings ending in b.

Since these are finite classes.. L is regular language

# Myhill Neorode Theorem

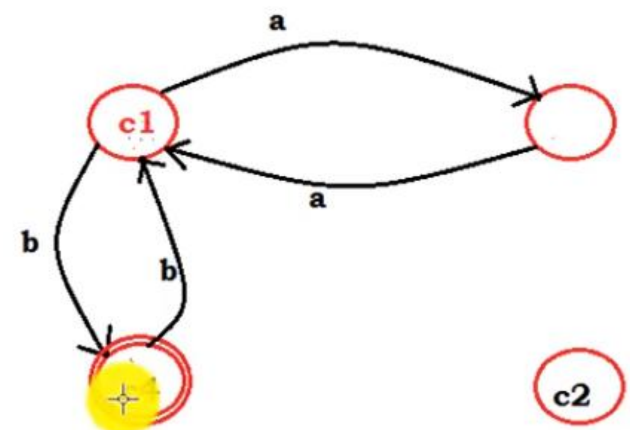
**Example#2:**

**Suppose L is EVEN EVEN language where $\Sigma = \{a,b\}$ In how many classes does L may partition $\Sigma^*$, explain briefly. Also state whether this language is regular or not.**

It can be observed that L partitions $\sum^*$ into the following classes:

~~C1 = no. of a is even and no. of b is odd~~
C2 = no. of b is even and no. of a is odd
C3 = no. of a is odd and no. of b is odd
C4 = no. of a is even and no. of b is even

hence the number of classes is finite which is 4.. so L is regular language...



# Myhill Neorode Theorem

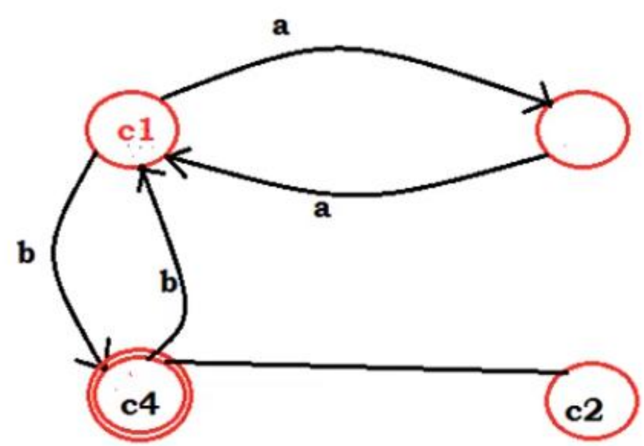
**Example#2:**

**Suppose L is EVEN EVEN language where $\Sigma$ = {a,b} In how many classes does L may partition $\Sigma^*$, explain briefly. Also state whether this language is regular or not.**

It can be observed that L partitions $\sum^*$ into the following classes:

~~C1 = no. of a is even and no. of b is odd~~
C2 = no. of b is even and no. of a is odd
C3 = no. of a is odd and no. of b is odd
C4 = no. of a is even and no. of b is even

hence the number of classes is finite which is 4.. so L is regular language...



# Myhill Neorode Theorem

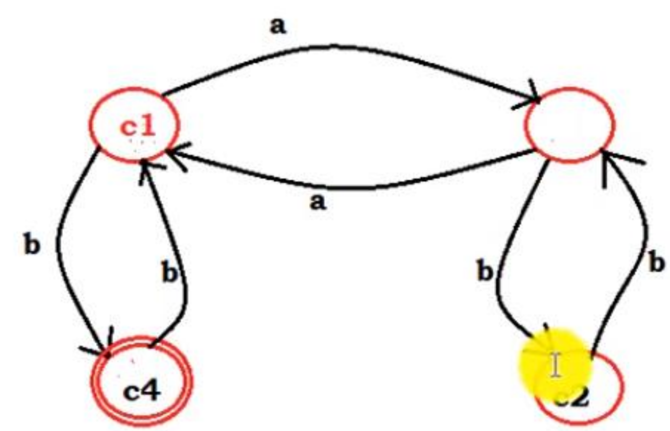
**Example#2:**

Suppose L is EVEN EVEN language where $\Sigma = \{a,b\}$ In how many classes does L may partition $\Sigma^*$, explain briefly. Also state whether this language is regular or not.

It can be observed that L partitions $\sum^*$ into the following classes:

~~C1 = no. of a is even and no. of b is odd~~
C2 = no. of b is even and no. of a is odd
C3 = no. of a is odd and no. of b is odd
C4 = no. of a is even and no. of b is even

hence the number of classes is finite which is 4.. so L is regular language...



# Myhill Neorode Theorem

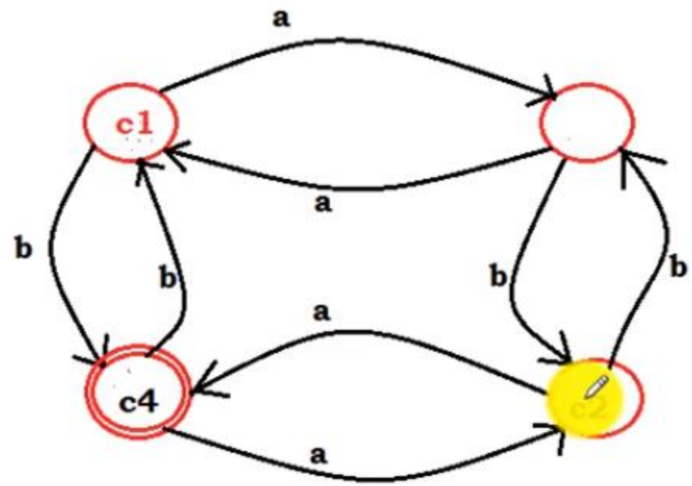
**Example#2:**

**Suppose L is EVEN EVEN language where Σ = {a,b}  In how many classes does L may partition Σ*, explain briefly.  Also state whether this language is regular or not.**

**It can be observed that L partitions ∑* into the following classes:**

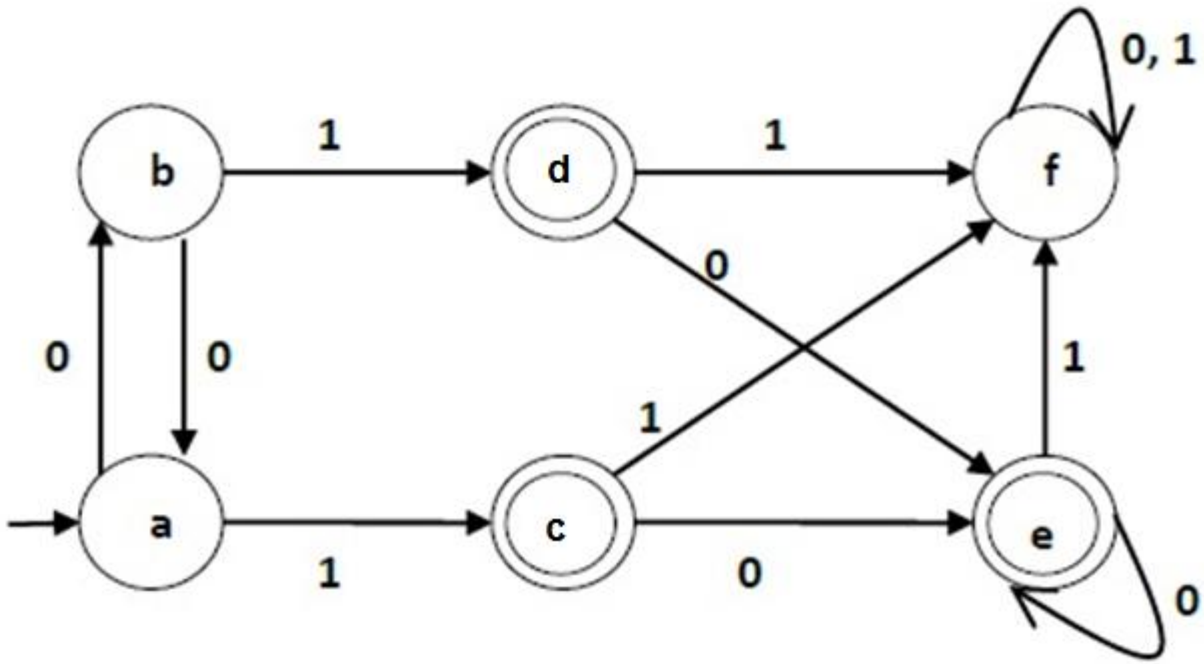
~~C1 = no. of a is even and no. of b is odd~~
C2 = no. of b is even and no. of a is odd
C3 = no. of a is odd and no. of b is odd
C4 = no. of a is even and no. of b is even

**hence the number of classes is finite which is 4.. so L is regular language...**



# DFA Minimization
# Using MyHill Nerode Approach

# Problem

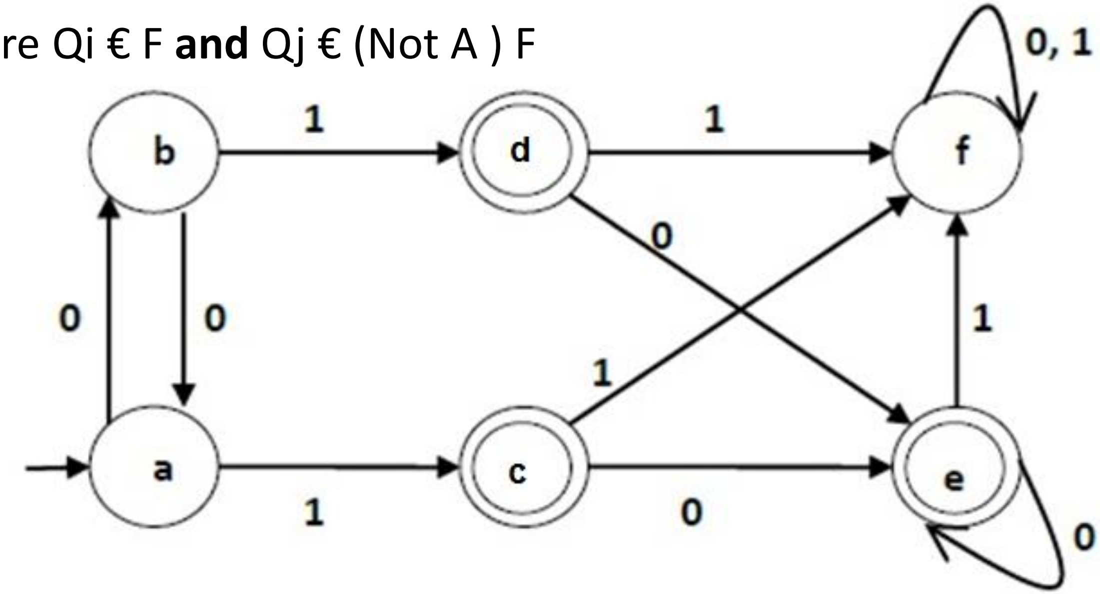


**Step 1**: Draw a table for all pairs of state of Qi and Qj

**Step 1** – We draw a table for all pair of states.

| | a | b | c | d | e | f |
|---|---|---|---|---|---|---|
| a | | | | | | |
| b | | | | | | |
| c | | | | | | |
| d | | | | | | |
| e | | | | | | |
| f | | | | | | |

**Step 2**: Mark or Tick all pairs where Qi € F **and** Qj € (Not A ) F

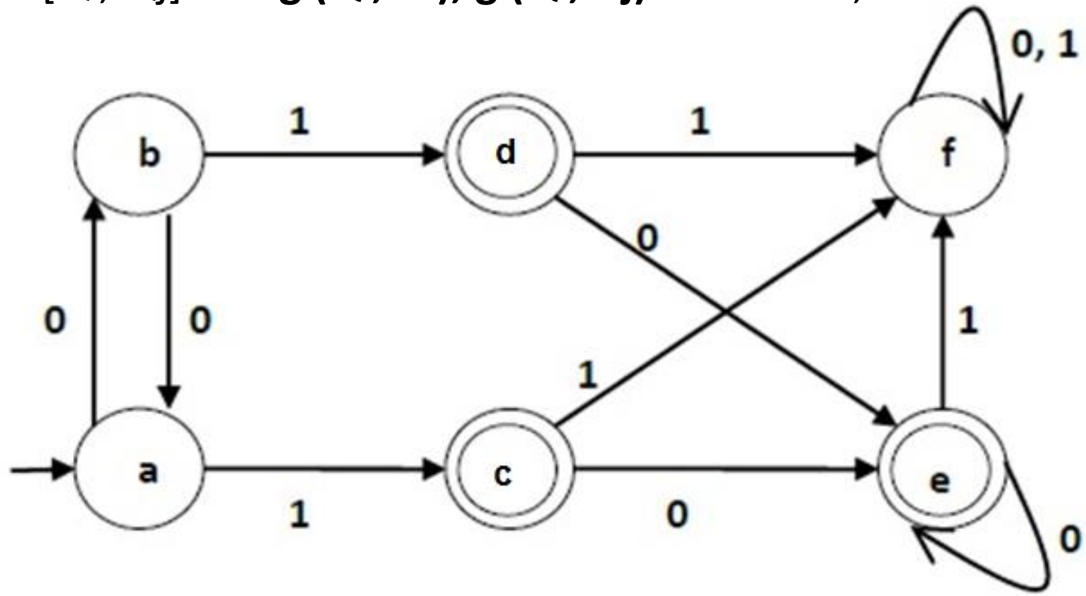



**Step 2** – We mark the state pairs.

| | a | b | c | d | e | f |
|---|---|---|---|---|---|---|
| a | | | | | | |
| b | | | | | | |
| c | ✓ | ✓ | | | | |
| d | ✓ | ✓ | | | | |
| e | ✓ | ✓ | | | | |
| f | | | ✓ | ✓ | ✓ | |

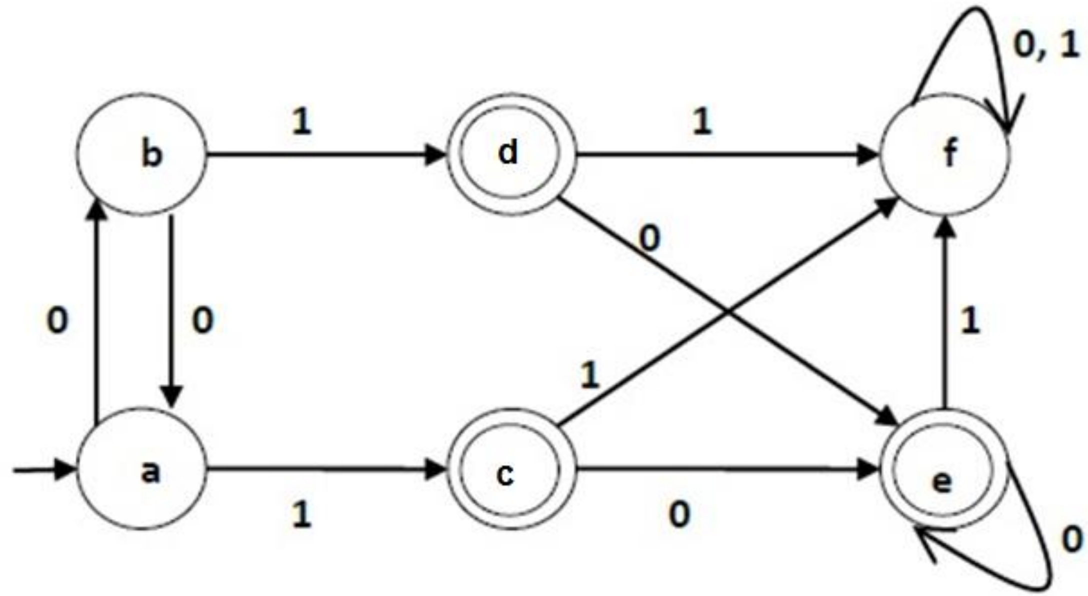
**Step 3**: If there is an unmarked pairs [Qi, Qj] and **ƺ (Qi, Ai), ƺ (Qi, Aj) is marked**, then mark [Qi, Qj]

**…repeat the step until we cannot mark anymore.**




| | a | b | c | d | e | f |
|---|---|---|---|---|---|---|
| a | | | | | | |
| b | | | | | | |
| c | ✓ | ✓ | | | | |
| d | ✓ | ✓ | | | | |
| e | ✓ | ✓ | | | | |
| f | ✓ | ✓ | ✓ | ✓ | ✓ | |

**Step 3**: If there is an unmarked pairs [Qi, Qj] and **ƺ (Qi, Ai), ƺ (Qi, Aj) is marked**, then mark [Qi, Qj]




**Step 3**: [a, b]

ƺ (a, 1) = c
ƺ (b, 1) = d

**C, d are not mark in previous table.**

**So no mark**

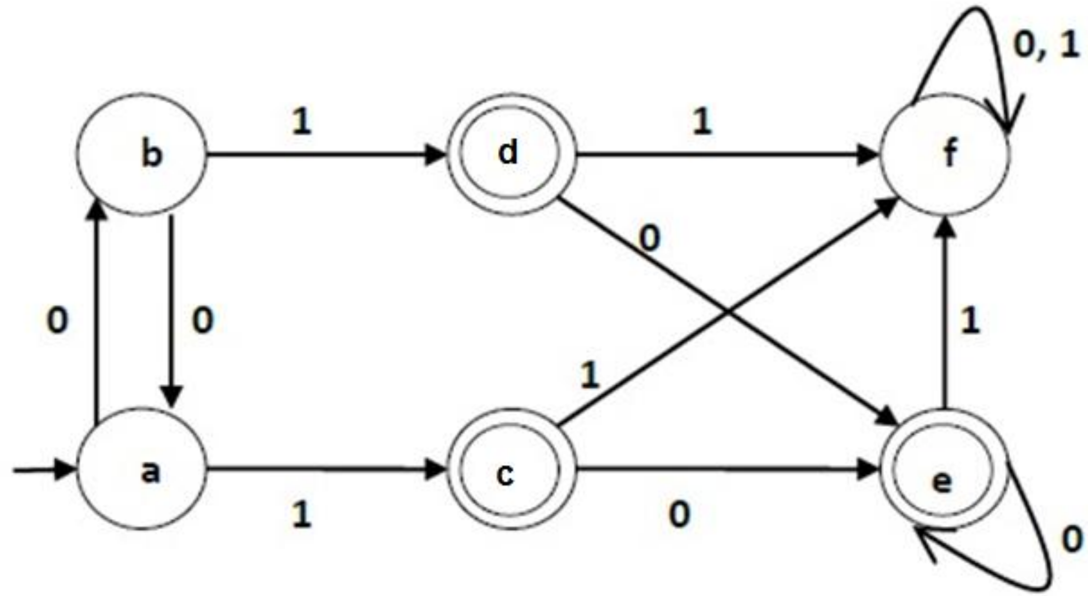
**Step 3**: [a, b]

ƺ (a, 0) = b
ƺ (b, 0) = a

A, b are not mark in previous table.

So no mark.

**Step 3**: If there is an unmarked pairs [Qi, Qj] and **ƺ (Qi, Ai), ƺ (Qi, Aj) is marked**, then mark [Qi, Qj]




**Step 3**: [d, c]

ƺ (d, 1) = f
ƺ (c, 1) = f

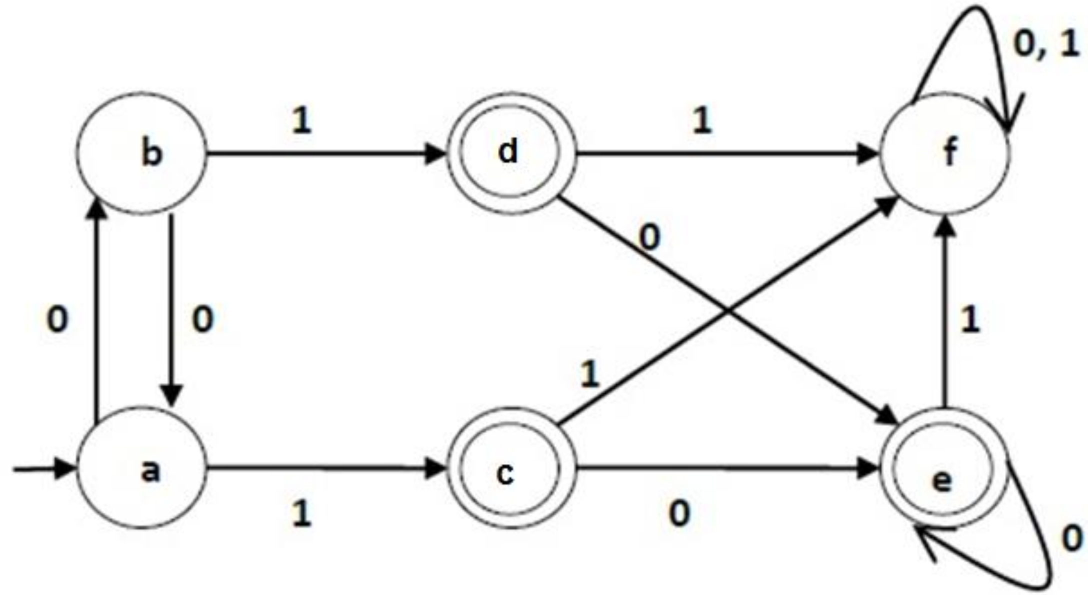
So no mark as no cell

**Step 3**: [d, c]

ƺ (d, 0) = e
ƺ (c, 0) = e

So no mark as no cell

**Step 3**: If there is an unmarked pairs [Qi, Qj] and **ƺ (Qi, Ai), ƺ (Qi, Aj) is marked**, then mark [Qi, Qj]




**Step 3**: [e, c]

ƺ (e, 1) = f
ƺ (c, 1) = f

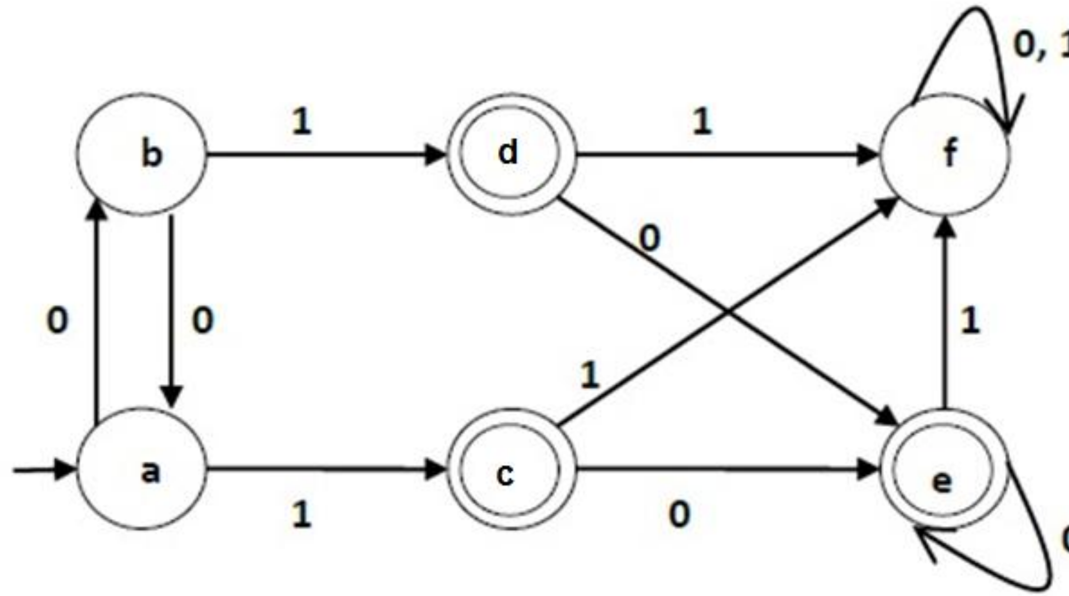
So no mark

**Step 3**: [e, c]

ƺ (e, 0) = e
ƺ (c, 0) = e

So no mark

**Step 3**: If there is an unmarked pairs [Qi, Qj] and **ƺ (Qi, Ai), ƺ (Qi, Aj) is marked**, then mark [Qi, Qj]




**Step 3**: [e, d]

ƺ (e, 1) = f
ƺ (d, 1) = f

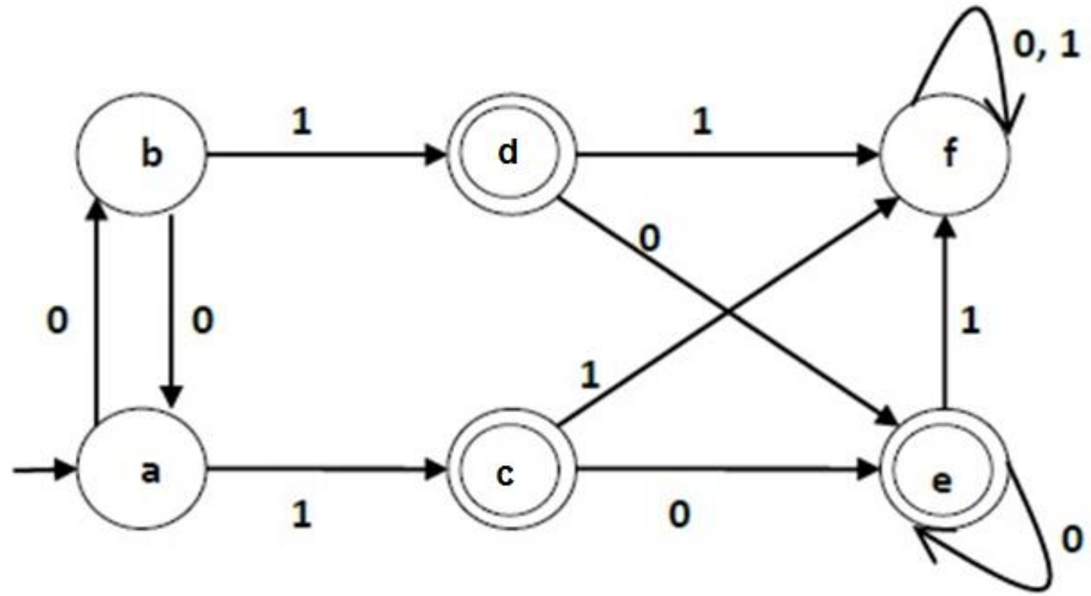
So no mark

**Step 3**: [e, d]

ƺ (e, 0) = e
ƺ (d, 0) = e

So no mark

**Step 3**: If there is an unmarked pairs [Qi, Qj] and **ƺ (Qi, Ai), ƺ (Qi, Aj) is marked**, then mark [Qi, Qj]




**Step 3**: [f, a]

ƺ (f, 1) = f
ƺ (a, 1) = c

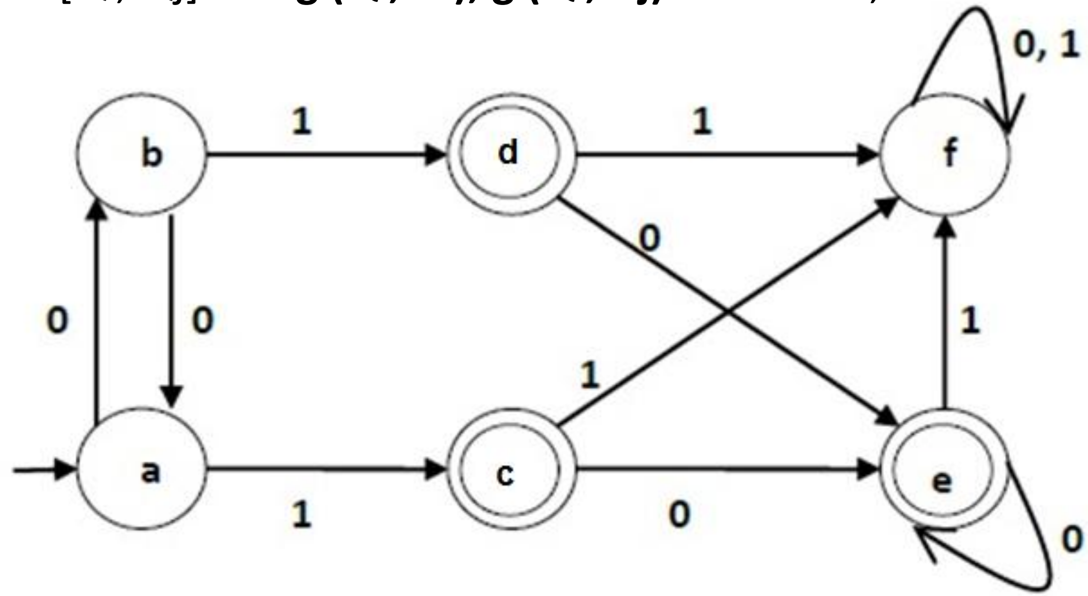
**So MARK**

**Step 3**: [f, a]

ƺ (f, 0) = f
ƺ (a, 0) = b

So no mark

**Step 3**: If there is an unmarked pairs [Qi, Qj] and **ƺ (Qi, Ai), ƺ (Qi, Aj) is marked**, then mark [Qi, Qj]




**Step 3**: [f, b]

ƺ (f, 1) = f
ƺ (b, 1) = b

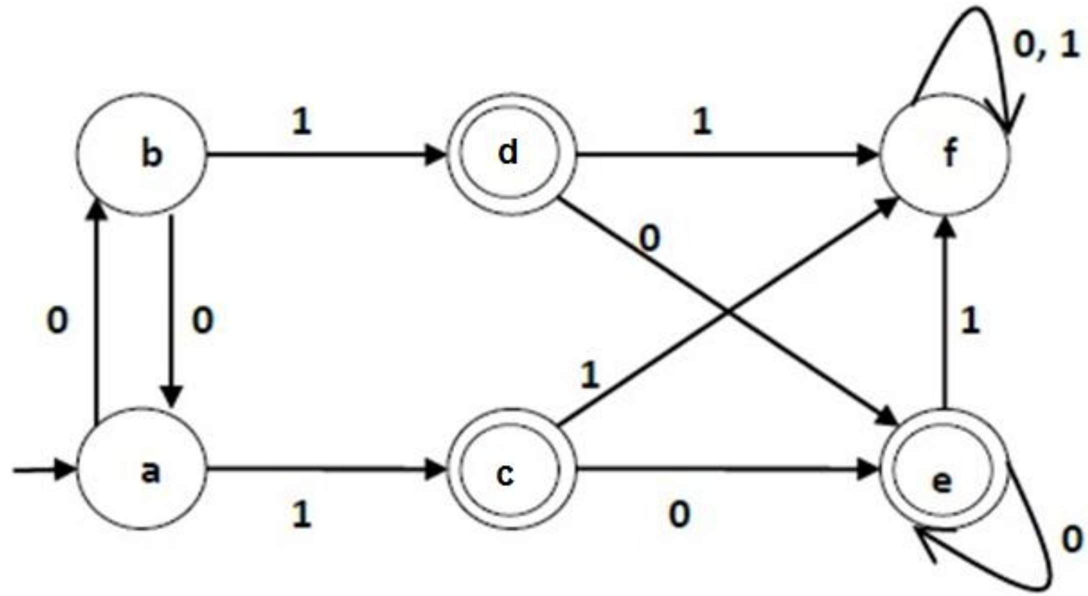
**So MARK**

**Step 3**: [f,b]

ƺ (f, 0) =
ƺ (a, 0) =

No need…

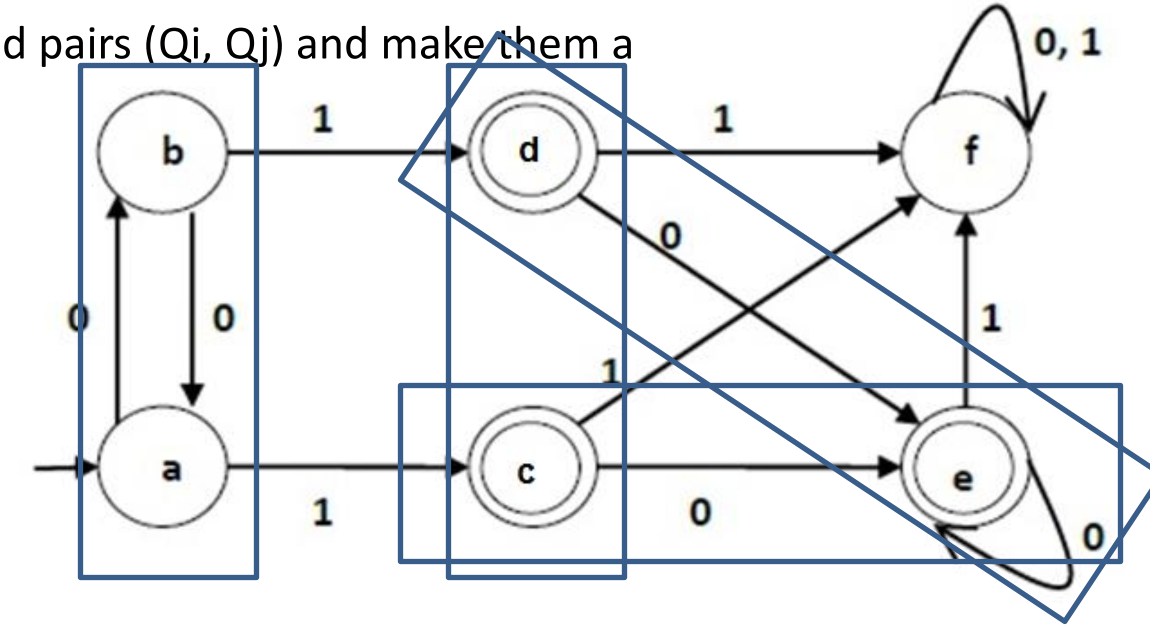
**Step 3**: If there is an unmarked pairs [Qi, Qj] and **ƺ (Qi, Ai), ƺ (Qi, Aj) is marked**, then mark [Qi, Qj]




**Step** :Repeat the steps until we cannot mark anymore …

**Step 4**: Combine all the unmarked pairs (Qi, Qj) and make them a single state in reduce DFA.

**Step 4**: (a,b), (d, c), (e, c), (e, d)




| | a | b | c | d | e | f |
|---|---|---|---|---|---|---|
| a | | | | | | |
| b | | | | | | |
| c | ✓ | ✓ | | | | |
| d | ✓ | ✓ | | | | |
| e | ✓ | ✓ | | | | |
| f | ✓ | ✓ | ✓ | ✓ | ✓ | |

**Step 4**: Combine all the unmarked pairs (Qi, Qj) and make them a single state in reduce DFA.

**Step 4**: (a,b), (d, c), (e, c), (e, d)