Minimization of DFA

- Minimization of DFA is required to obtain the minimal version of any DFA which consists of the minimum number of states possible.
- A DFA designed with 5 states and another DFA designed with 4 states, both doing the same task. Here both are correct but the DFA with 4 states is more efficient.
- Sometimes it might be difficult to design a DFA directly with the minimum number of states.
- To minimize a DFA, we need to combine two states into one but it is possible when those two states are equivalent.

Two states are said to be equivalent if –

$\delta(A, X) \rightarrow F$ and $\delta(B, X) \rightarrow F$

OR where X is any input string.

$\delta(A, X) \not\rightarrow F$ and $\delta(B, X) \not\rightarrow F$

Different types of equivalences:

If **|X| = 0**, then A and B are said to be **0 equivalent**.

If **|X| = 1**, then A and B are said to be **1 equivalent**.

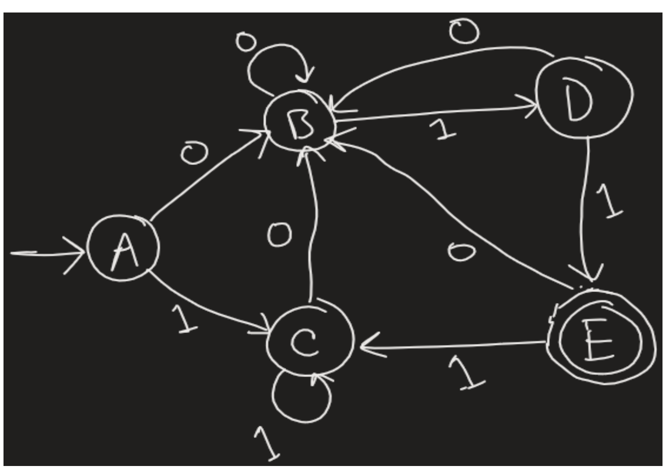
If **|X| = 2**, then A and B are said to be **2 equivalent**.

.

.

If **|X| = n**, then A and B are said to be **n equivalent**.

Example:



Transition Table:

	0	1
→A	B	C
B	B	D
C	B	C
D	B	E
*E	B	C

0 Equivalence: {A, B, C, D} {E}

1 Equivalence: {A, B, C} {D} {E}

2 Equivalence: {A, C} {B} {D} {E}

3 Equivalence: {A, C} {B} {D} {E}

Minimal DFA: