Given that *B* and *C* are two languages and $B \xleftarrow{1} C = \{w \in B \mid \text{ for some } y \in C, \text{srings } w \text{ and } y \text{ contain equal numbers of } 1\}$ over the alphabet $\Sigma = \{0,1\}$

We have to prove that class of regular languages closed under $\xleftarrow{1}$ operation

That means if *B* and *C* are regular languages than $B \xleftarrow{1} C$ is also a regular language.

So given that *B* and *C* are regular languages.

We know that

"A language is regular if it is recognized by an automation"

• Let M_B be the DFA that recognizes the language *B*

$M_B = (Q_B, \Sigma, \delta_B, q_B, F_B)$

• Let M_c be the DFA that recognizes the language *C* $M_C = (Q_C, \Sigma, \delta_C, q_C, F_C)$

Now we have to construct an NFA which recognizes $B \xleftarrow{1} C$.

<u>Construction of NFA to recognize $B \xleftarrow{1} C$:</u>

Let $N = (Q, \Sigma, \delta, q_0, F)$ be the NFA.

Now *N* has to decide whether a string $w \in B \xleftarrow{1} C$ or not.

• For that first machine M checks whether $w \in B$ or not.

• If $w \in B$, then non deterministically find out a string of that contains the same number of 1s as contained in *w* and checks that $y \in C$.

• That means for each string *B*, there are $C(\text{number ofstrings in } C)$ parallel machings will exist

Thus $Q =$ set of states

$= Q_B \times Q_C$

$\Sigma =$ set of alphabet

= same as B and C

δ is given by, for $(q,r) \in Q$ and $a \in \Sigma$

$$\delta(q,r),a = \begin{cases} \{(\delta_B(q,0),r)\} & \text{if } a = 0 \\ \{((\delta_B(q,1)),\delta_C(r,1))\} & \text{if } a = 1 \\ \{(q,\delta_C(r,0))\} & \text{if } a = \in \end{cases}$$

$q_0 =$ start state

$= (q_B, q_c)$

$F =$ set of final states

$= F_B \times F_C$

Thus we defined an NFA *N* to recognize $B \xleftarrow{1} C$.

Hence $B \xleftarrow{1} C$ is regular.

Therefore class of regular languages closed under $B \xleftarrow{1} C$ operation