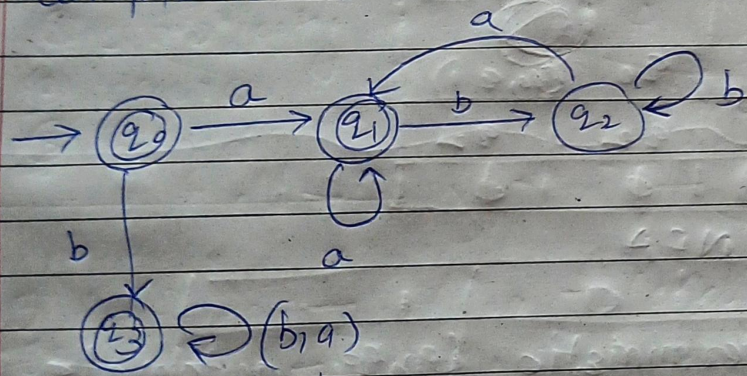


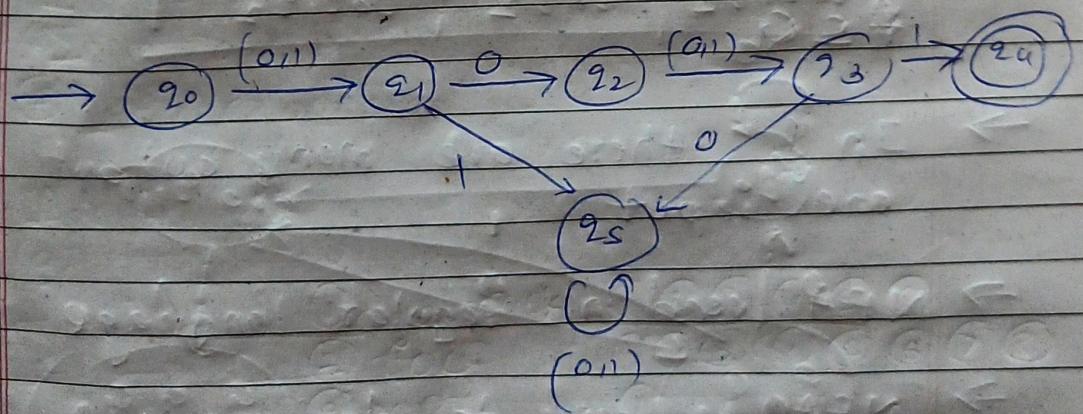
$L = \{ab, aab, a^2bab, a^3\}$
 a b a b
 a

$\Rightarrow L = \{ \text{not starting with } a, \text{ not ending with } y \}$

Complement



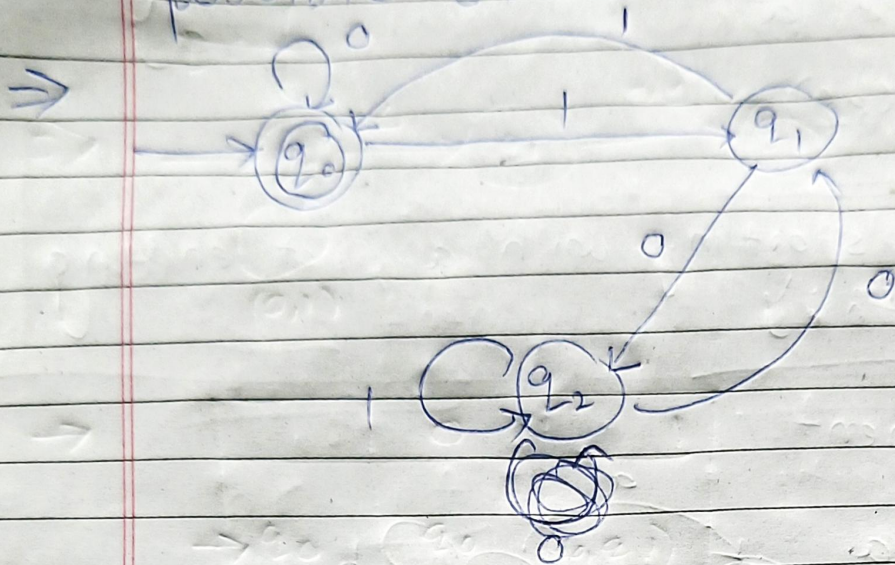
$\Rightarrow L \{ \text{second symbol is } 0 \text{ \& } 4^{\text{th}} \text{ is } 1 \}$



$\Rightarrow L = \{ \text{binary string divisible by } 3 \}$

remainder $\rightarrow 0, 1, 2$

possible remainder $\rightarrow 3 \rightarrow 0, 1, 2$
possible states $\rightarrow 3$



* NDEFA / NFA

\rightarrow Non deterministic finite automata

\rightarrow NFA $(Q, \Sigma, q_0, f, \delta)$

$f \subseteq Q$

\rightarrow In f there are many output for one input

\rightarrow NFA used in regular language

\rightarrow NFA is easy to design

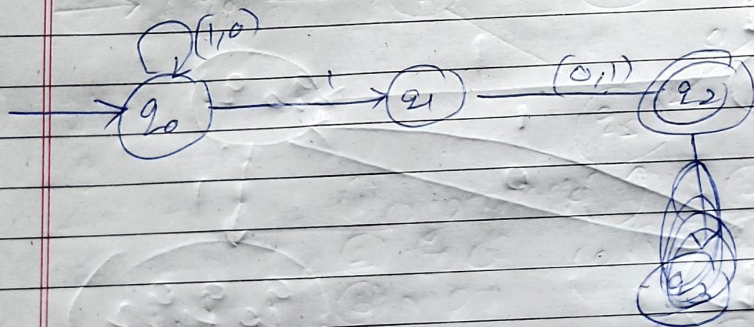
* DFA vs NFA

① dead config not allowed	① dead config allowed
② multiple choices not allowed	② multiple choices allowed
③ ϵ -move not allowed	③ ϵ move allowed
④ deterministic	④ Not deterministic
⑤ designing difficult	⑤ designing is easy

* e.g

$L = \{ \text{binary strings 2}^{\text{nd}} \text{ last is 1} \}$

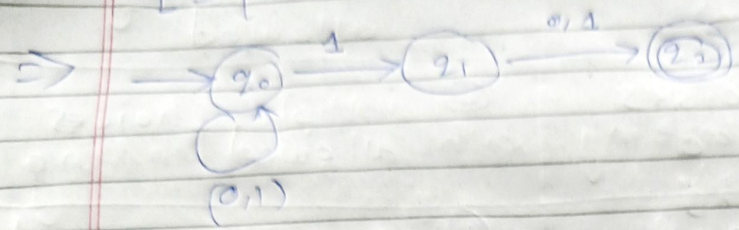
$L = \{ .10, 010, 0110, 00000010, \dots, 01010110 \}$



$L = (0+1)^* (1) (0+1)^*$

* Converting NFA to DFA

Ex -



→

	0	1
→ q ₀	q ₀	(q ₀ , q ₁)
q ₁	q ₂	q ₂
(q ₂)	∅	∅

→

	0	1
q ₀	q ₀	q ₀ q ₁
q ₀ q ₁	q ₀ q ₂	q ₀ q ₁ q ₂
q ₀ q ₂	q ₀	q ₀ q ₁
q ₀ q ₁ q ₂	q ₀ q ₂	q ₀ q ₁ q ₂

