Date:

$$(Q, \underset{}{\overset{\text{input alphabet}}{\Sigma}}, \underbrace{T}_{\text{tape alphabet}}, \delta, q_0, F)$$

DTM

$$\delta : Q \times T \rightarrow Q \times T \times \{\overset{\text{left or right.}}{L, R}\}$$

NTM

$$\delta : Q \times (T \cup \{\varepsilon\}) \rightarrow Q \times (T \cup \{\varepsilon\}) \times (L, R)$$

Here input comes on tape so.

$$\Sigma \subset T$$

a	a	b	b	c	c	α	β	⊔	⊔	⊔

input. some other tape alphabets. blank spaces

Acceptance criteria:

halt → accept → accepted even though whole input isn't read.

halt → reject → reject state.

may not halt.

eg $L = \{ w\#w \mid w \in \{0,1\}^* \}$ — $*$: 0 or more repeatations.

(# called sharp).

00110 #00110 ✓

011 #01 ✗

$\delta : Q \times T \rightarrow Q \times T \times \{L, R\}$

α B γ

reads α from tape — writes this — goes to right or left

effective transitions will be of form

$$\bigcirc \xrightarrow{\alpha \rightarrow B, \gamma} \bigcirc$$

Let us notice one iteration of comparision

- We label the verified ones with x.

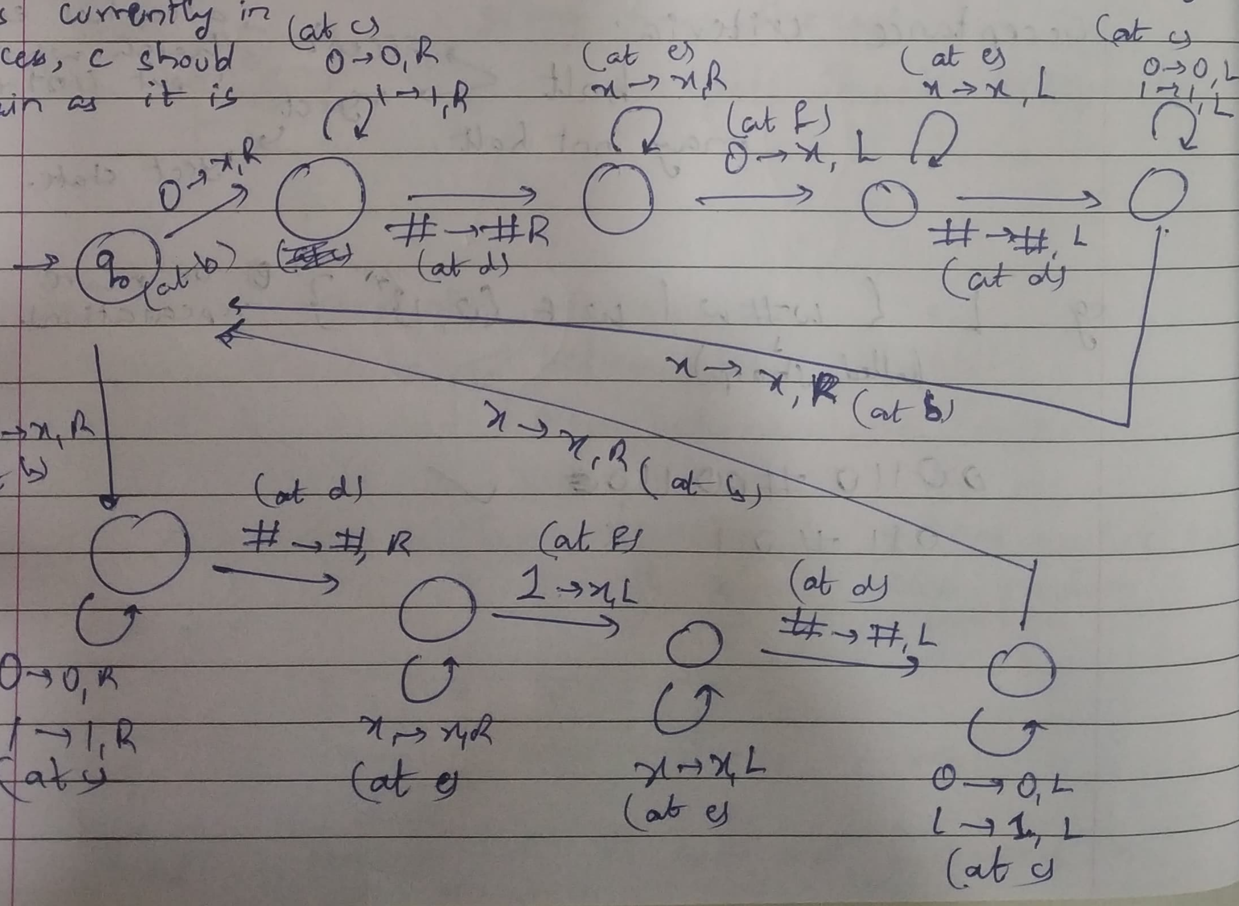
$\underbrace{0110}\ \square \ldots\ \#\ \cancel{00}\ \underbrace{0110}\ \square \ldots$

~~let~~ suppose, these are matched and we replace x with them on tape

- Labelled parts of tape for better understanding.

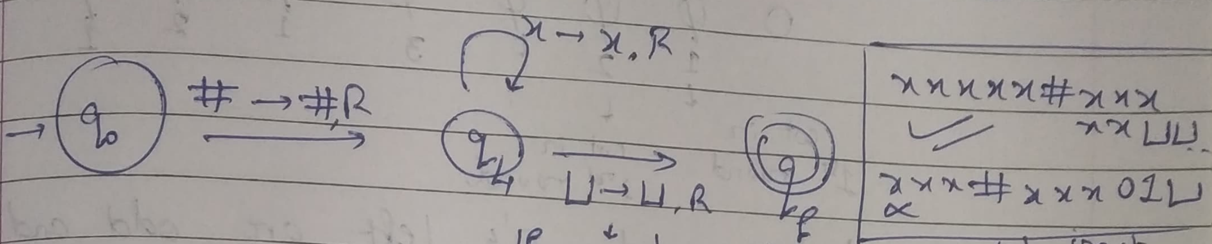
$\underbrace{xxxx}_{a}\ \underbrace{\square}_{b}\ \underbrace{\ldots}_{c}\ \underbrace{\#}_{d}\ \underbrace{xxxx}_{e}\ \underbrace{\square}_{f}\ \underbrace{\ldots}_{g}$

- all a's are processed, b is currently in process, c should remain as it is



This machine writes two ~~so~~ strings simultaneously

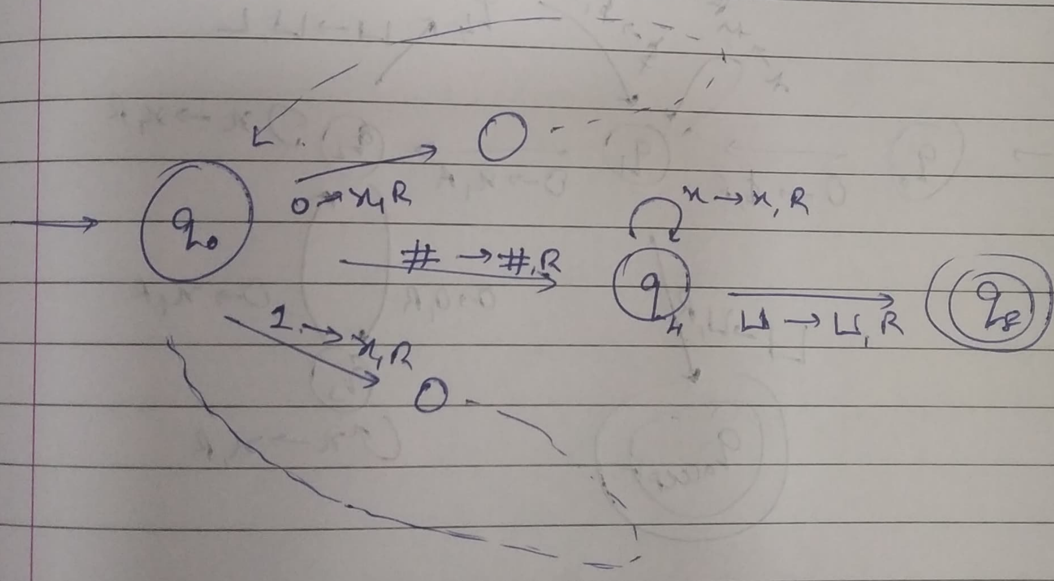
If we have 011 # 0111 then this machine will stay on starting state. ~~but~~ we must reject this, or make correct one to accept, so one branch is added to it.



If we have blank, then only input will be accepted

Due to limitation of space i have drawn them ~~separteate~~ separate.

rough diagram of whole machine



eg. $L = \{ 0^{2^n} \mid n \geq 0 \}$

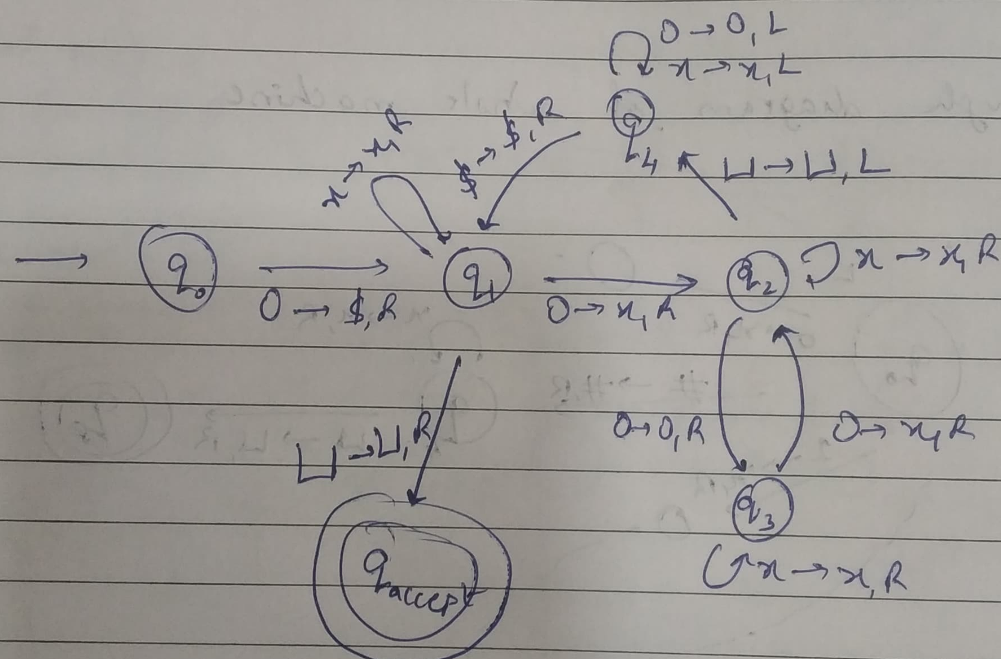
$0^{2^0} = 0^1 = 0$
$0^{2^1} = 0^2 = 00$
$0^{2^2} = 0^4 = 0000$
$0^{2^3} = 0^8 = \ldots$

logic / idea to build a machine

0 $\cancel{0}$ $\cancel{0}$ $\cancel{0}$ $\cancel{0}$ $\cancel{0}$ $\cancel{0}$ $\cancel{0}$
 1 2 1 3 1 2 1

1 ↓ cut in 1st round
2 ↓ cut in 2nd round

In the last round, 0's left are odd and equal to 1, and in rest rounds, there are even 0's left.



This machine on its own rejects odd input strings.

$\cancel{0}$ $\cancel{0}$ 0 $\cancel{0}$ 0 → q_3 machine will halt in q_3

We make $\sqcup \to \sqcup, R$ transition, to be safe that whole input is processed