TURING MACHINE

9- tuple

$M = (Q, \Sigma, \Gamma, \vdash, \sqcup, \delta, s, t, r)$

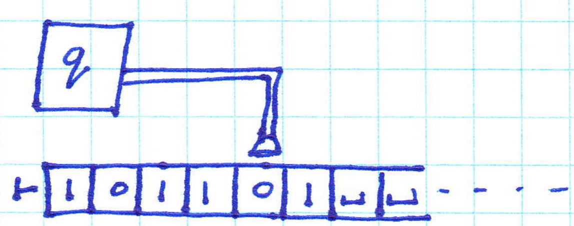
Q: finite set of states
Σ: a finite set of input symbols
Γ: a finite set of tape symbols $\Gamma \supset \Sigma$
$\sqcup \in \Gamma \setminus \Sigma$ the blank symbol
$\vdash \in \Gamma \setminus \Sigma$ the end marker - left
δ: $Q \times \Gamma \longrightarrow Q \times \Gamma \times \{L, R\}$ transition function
$s \in Q$ start state
$a \in Q$ accept state
$r \in Q$ reject state

$\delta(q, a) = (q', b, L)$ means if the machine reads a & is in state q, it changes state to q', erases the a & writes b and moves one step to the <u>left</u>.

TM's can move left or right on the tape
You cannot overwrite the left end marker
You cannot move left of $\vdash$
Once it enters a or r it never leaves

$\delta(q,a) = (q', a, L)$ means you leave the symbol unchanged.

A <u>CONFIGURATION</u> is a description & the machine at an instant of time



$1\,0\,1\,1\,q\,0\,1 \rightarrow$ HOW TO WRITE THE CONFIGURATION AS A STRING

$u\,a\,q_i\,b\,v$ <u>YIELDS</u> $u\,q_j\,a\,c\,v$
if $\delta(q_i, b) = (q_j, c, L)$

Given M & input ω the

<u>start</u> configuration is $q_0\,\omega$ or $s\,\omega$

an accept configuration is any configuration in which the state is a or q_a, similarly for r or q_r. An accept or reject configuration is called a <u>halting</u> configuration

M <u>accepts</u> ω if there is a <u>finite</u> <u>sequence</u> of configurations $C_1, C_2, \cdots C_k$ s.t.

1. C_1 is the start configuration $s\omega$
2. Each C_i yields C_{i+1}
3. C_k is an accepting configuration

$L(M) = \{ w \in \Sigma^* \mid M \text{ accepts } \omega \}$

3 outcomes are possible : accept, reject, loop forever

L is Turing recognizable if $\exists$ TM M s.t $L = L(M)$

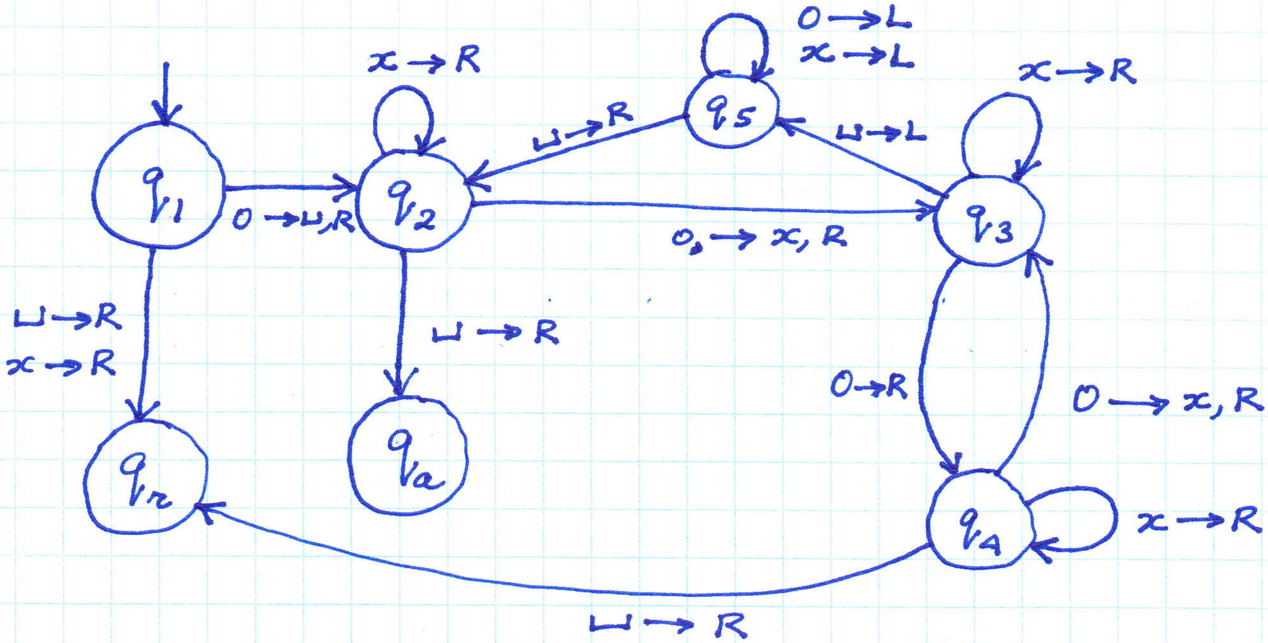
L is Turing decidable if $\exists$ TM M s.t. $L = L(M)$ & $\forall w \in \Sigma^*$ M <u>halts</u> on ω.

We say L is computably enumerable or CE if $L = L(M)$ for some TM

We say L is computable or decidable if M always halts & $L = L(M)$

Old terminology : recursively enumerable (RE) for CE & recursive for decidable.

Obviously any decidable language is recognizable (CE).



$\Sigma = \{0\}$ $\quad\quad$ $\Gamma = \{0, \sqcup, x\}$

$q_1 \xrightarrow{0 \to \sqcup, R} q_2$ means: read 0, replace it with $\sqcup$, move right 1 cell & change state from q_1 to q_2.

$q_3 \xrightarrow{0 \to R} q_4$ means: read 0, don't change it move right one cell & change state from q_3 to q_4

This machine recognizes $\{0^{2^n} \mid n \geq 0\}$

Idea: cross off (replace 0 with x) every 2^{nd} 0

first 0 is replaced by $\sqcup$ to serve as left endmarker

If there are an odd number of 0's at any phase reject