Lecture 10:

* checkup 3 => reopened until the 13th

* HW 2 => due Monday 11:59 pm
(remember you have a late hour budget)

* No checkup 4 yet...

* HW 1 regrade requests

* Quiz 1: we're grading it manually
that's what counts

also Grady for feedback

Q5 was a "bonus", we didn't expect you to solve it.
(we graded more conservatively, no IDK)

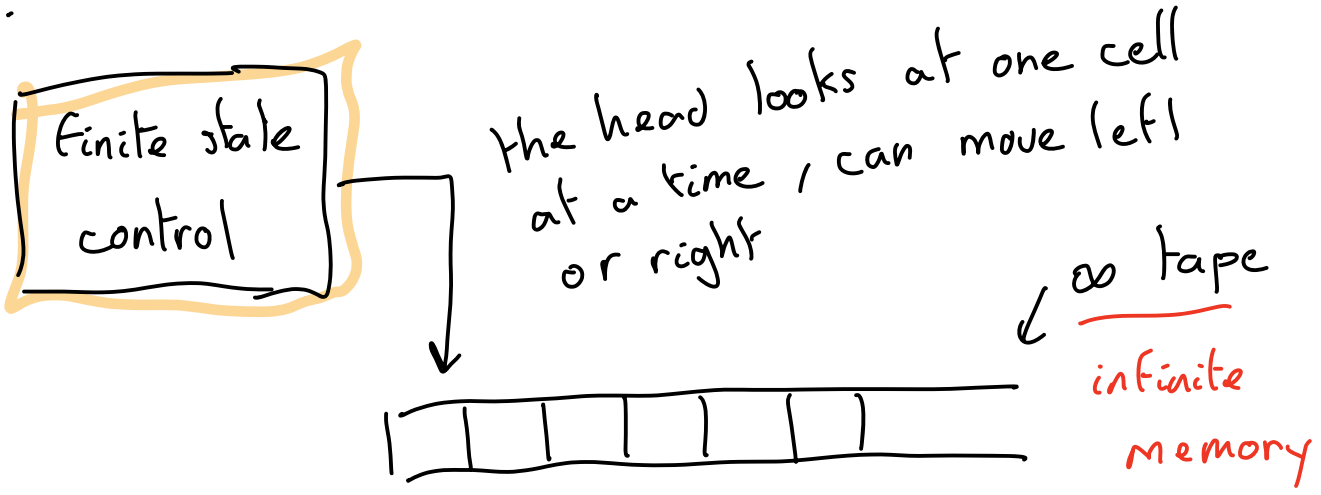
Now: Turing Machines

not like DFA

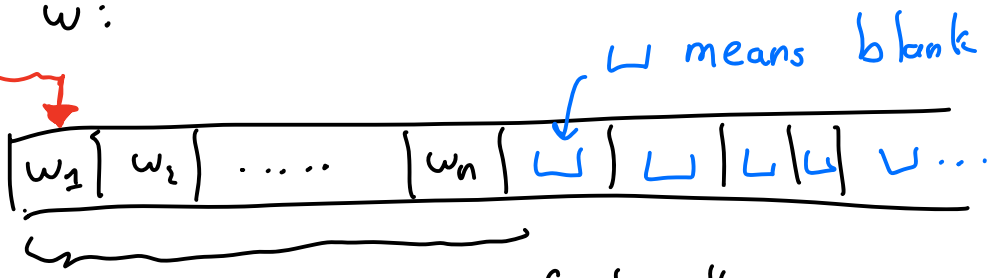
- unbounded memory
- unlimited access to the input
  (we can read the same parts of the input many times)
- write to memory

- they still output accept/reject (or nothing)
  → we will not really talk about computing functions / richer outputs

TM:



On input w:



A computation of a TM on input w:

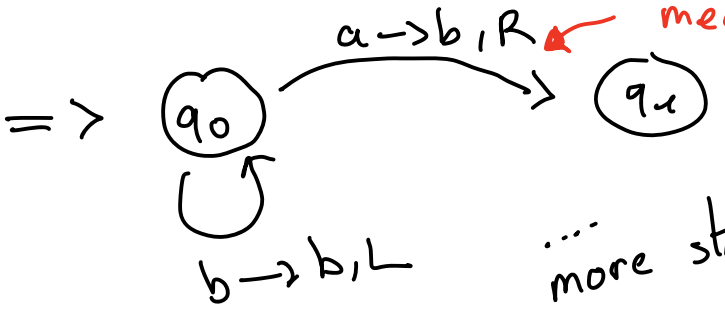
start configuration:
- start in q_0 (the start state)
- head points to the left most cell of the tape
- tape contains w followed by ∞ many "␣" blank symbols.

each step: depending on:
- current state
- current symbol under the head

do:
- switch state
- write on tape (overwrite current symbol)
- move head left or right.

- two special states q_{acc}, q_{rej}
  as soon as you get into q_{acc} or q_{rej}: halt

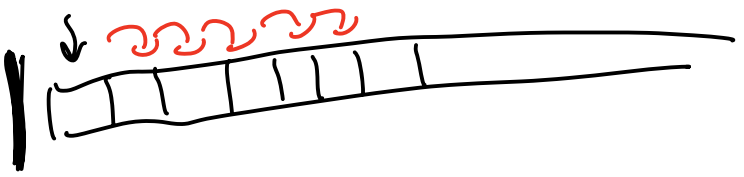
state diagram:



means if current state is q_0, and the head reads "a" then:
- replace "a" by "b" on the current cell
- move head right
- change state to q_1

if current state is q_0 and you read "b":
- replace "b" by "b"
- move the head left
- current state stays q_0

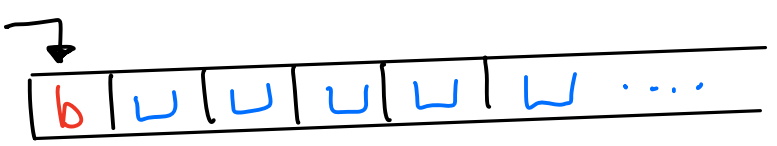
Note:



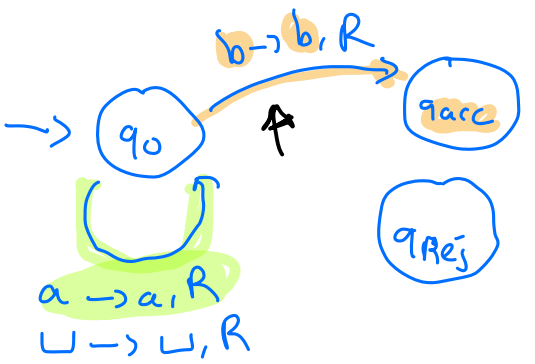
If the head is in the left most cell and we try to move left: the head does not move.

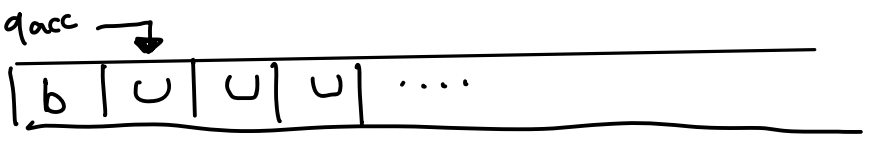
Example:

Computation on 'b':



M:

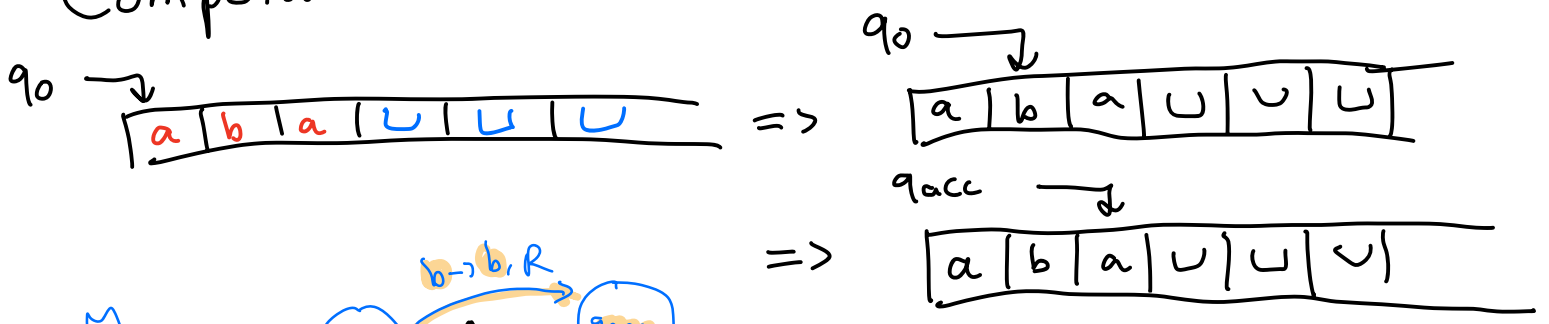




The TM accepts

$L(M) = \{$ all strings that contain a "b" $\}$

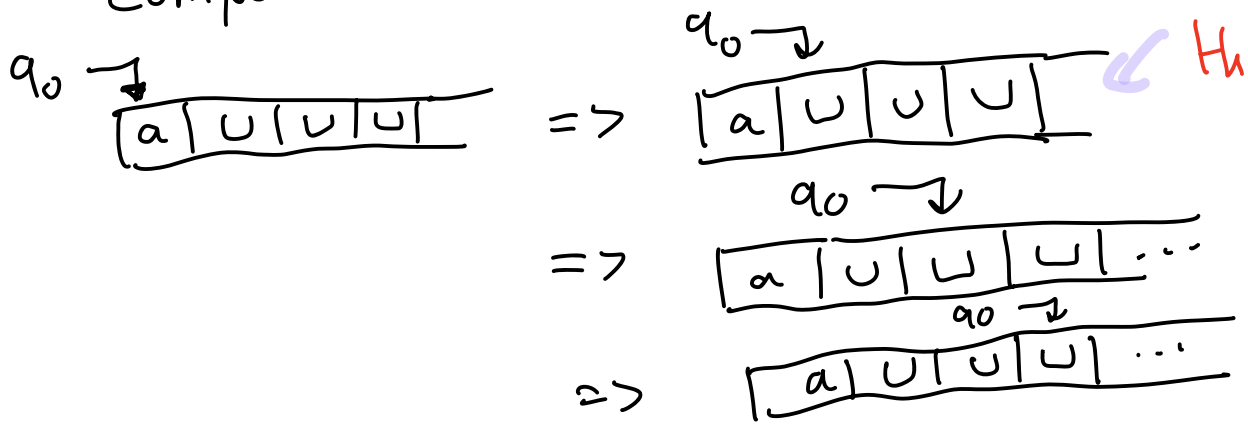
Computation on 'aba'





so we accept.

computation on 'a'

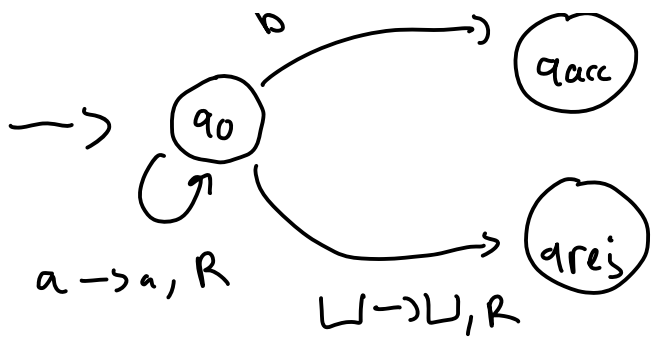


the new TM would Reject

The turing machine does not halt.

If the input contains a "b" the TM accepts otherwise it runs forever.

b, R



accepts if input has a 'b' and rejects otherwise

"go over the tape from left to right: if you ever see a "b" you accept, if you see "␣" (you read all the input) you reject"

Differences between TM and DFAs:

- can go left and right on the input
- unbounded memory
- can read and write
- immediately halt when you enter q_{acc} or q_{rej}

Def: A Turing Machine (TM) is a 7-tuple

$M = (Q, \Sigma, \Gamma, \delta, q_0, q_{acc}, q_{rej})$

- Q: a finite set of states
- Σ: input alphabet
- Γ: the "tape alphabet":
  - $\Sigma \subseteq \Gamma$ ~> $\Sigma = \{a, b\}$
  - $\sqcup \in \Gamma$ ~> $\Gamma = \{a, b, \sqcup\}$
  - $\sqcup \notin \Sigma$

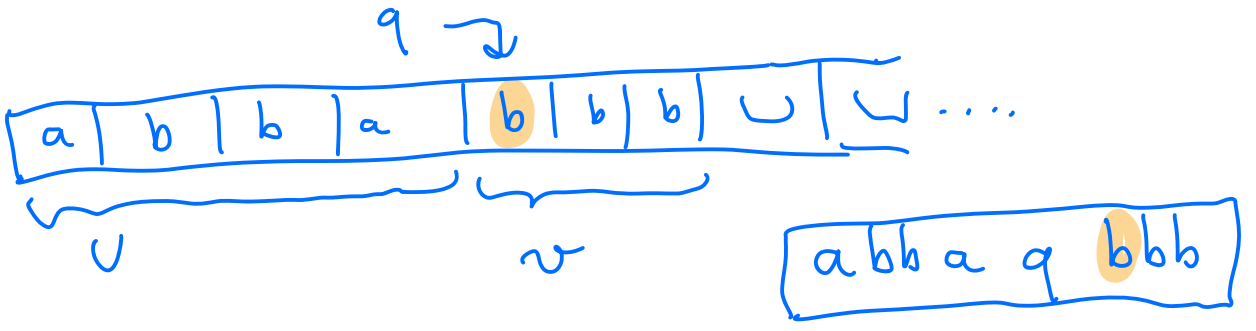
- $q_0, q_{acc}, q_{rej} \Rightarrow$ q_0 is "start state"
  q_{acc} is "accept state"
  q_{Rej} is "Reject state"

- $\delta : Q \times \Gamma \longrightarrow Q \times \Gamma \times \{L, R\}$

  Q: current state; Γ: current symbol under the head; Q: new state; Γ: what do you write; $\{L, R\}$: left / right

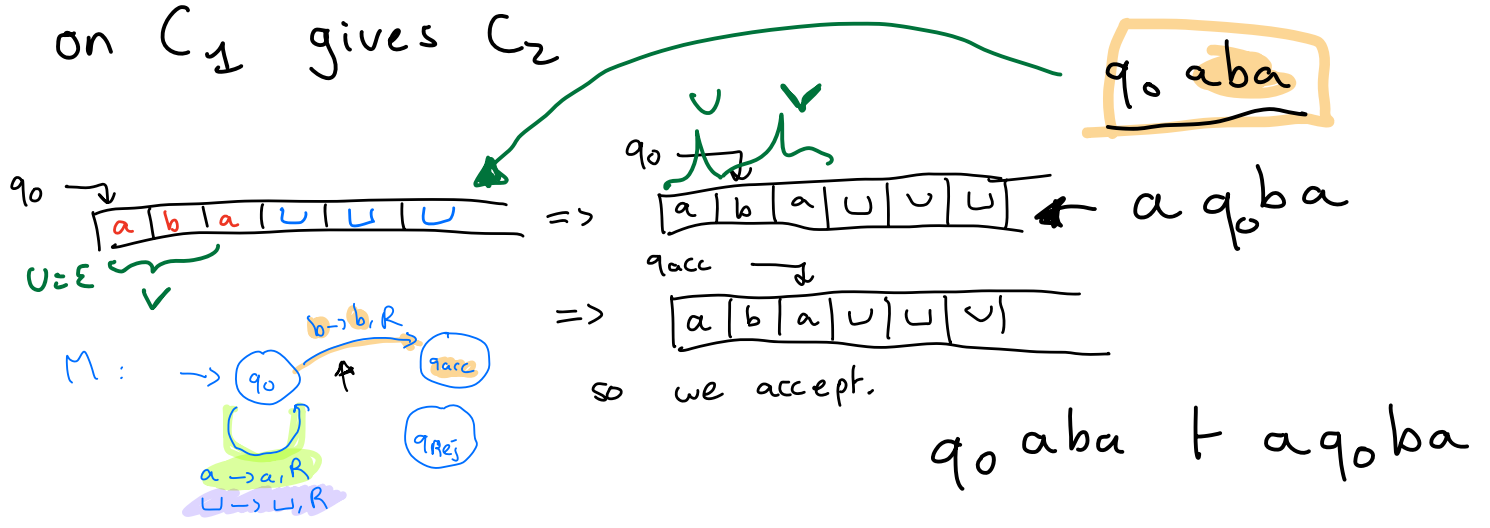
Def: A configuration of a TM M is of the form uqv where $u, v \in \Gamma^*$, $q \in Q$

This represents the current status of the computation.
- content of the tape is uv followed by $\sqcup$
- state is q
- the head points to the first symbol of v



For two configurations C_1, C_2 of M we say

$C_1 \vdash C_2$ ("C_1 yields C_2") if applying δ on C_1 gives C_2



q_0 aba $\vdash$ a q_0 ba

Def: for a TM $M = (Q, \Sigma, \Gamma, \delta, q_0, q_{acc}, q_{Rej})$ and a string $w \in \Sigma^*$:

- initial configuration of M on w is $q_0 w$
- for any $u, v \in \Sigma^*$
  - $u\, q_{acc}\, v$ is an accepting configuration
  - $u\, q_{rej}\, v$ is a rejecting configuration

M accepts w if there is a sequence of configurations $C_1, C_2, \dots C_m$ such that:

- $C_1 = q_0 w$ (start configuration)
- $C_i \vdash C_{i+1}$ for all $i \in \{1, \dots, m-1\}$
- $C_m = u\, q_{acc}\, v$ for some $u, v \in \Gamma^*$

(you end in accepting configuration)

Def: The language recognized by a TM M is

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

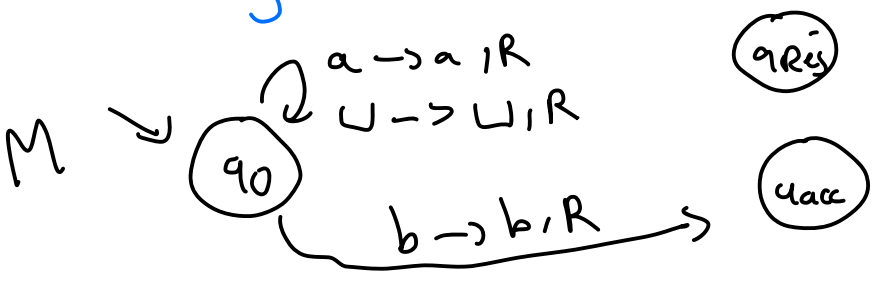
If a TM doesn't accept w $\Rightarrow$ it doesn't mean it rejects w
(M could run forever)

On input w: 3 options
- accept
- reject
- loop (the TM doesn't stop)

Def: A language $L \subseteq \Sigma^*$ is **TM-recognizable** ("recursively enumerable")
if $\exists$ a TM M that recognizes L
(so $L(M) = L$)

Def: A TM that halts (enters q_{acc} or q_{Rej}) on every input $w \in \Sigma^*$ is called a **decider**

A language L is **TM-decidable** if $\exists$ a TM M that recognizes L (so $L(M) = L$) and M is a decider



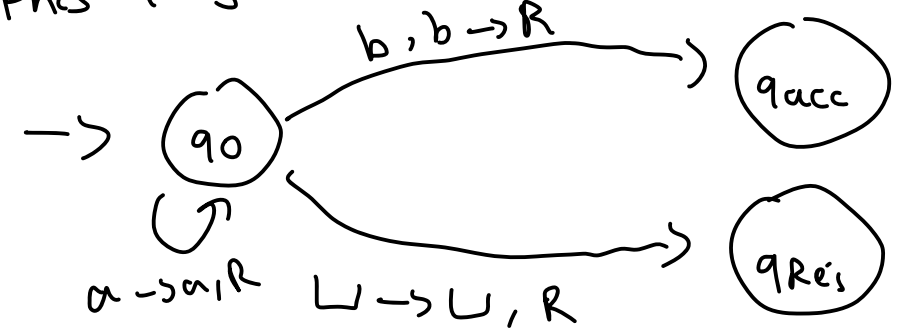
is this a decider?

No! this TM

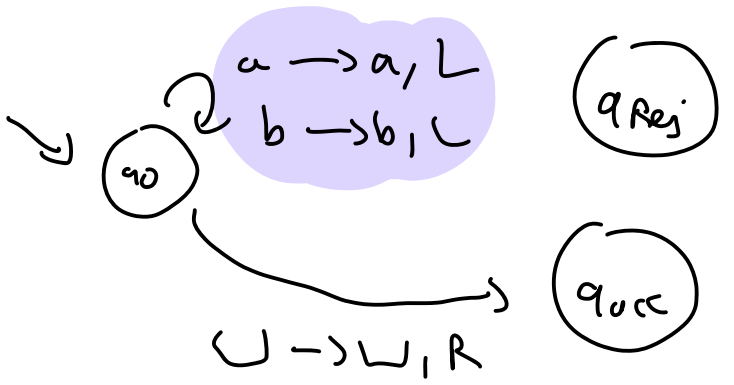
Runs forever on "a"

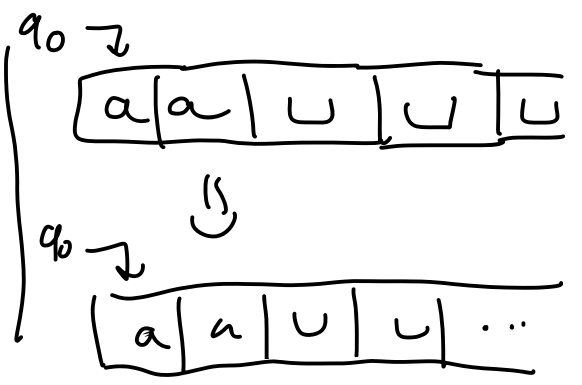
$L(M) = \{$all strings that contain a "b"$\}$

↑ this language is TM-recognizable.



this is a decider with the same language





$L(M) = \{\varepsilon\}$

↑ we gave a recognizer

is this decidable?

