CPSC 421/501 Oct 28, 2021

- Problem 6.1.6(a) requires $n \geq n_0$ (explained on solutions to Homework 5) ← thanks to Kehong

- Homework 6 solutions and some midterm practice to appear tomorrow (Friday). Brief solutions to some will appear Monday.

- Last 1/2 of Tuesday's class I'll take questions on midterm practice.

- Problem 1 on Individual Homework 5 will not be collected; a partial solution is given. [You will want to be sure to know how to do these types of problems when studying for the midterm and/or final.]

Last time:

Introduced TM's (Turing machines).

Goals: ① P = polynomial time on a TM (Turing machine) is a reasonable model for poly time as in CPSC 320, etc.

② One can build a universal Turing machine

=

Start small and build up

[Sip]: TM = DFA with

some extra powers:

Formally

move tape head

new state

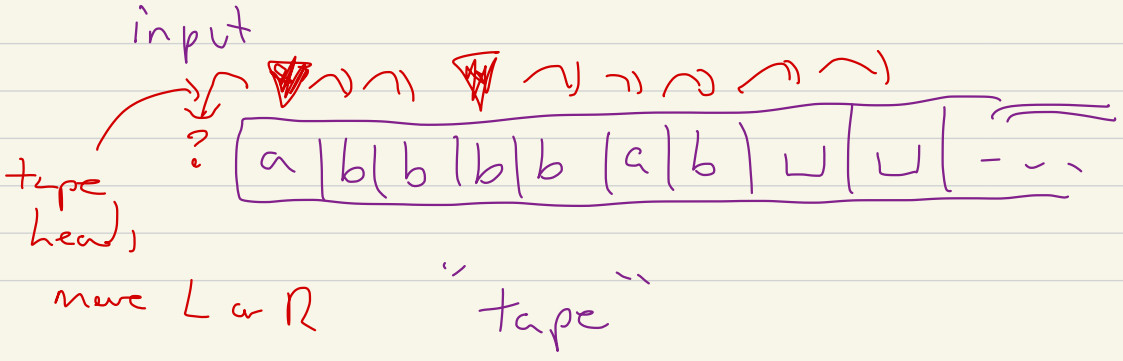
$$\delta : Q \times \Gamma \longrightarrow Q \times \Gamma \times \{L, R\}$$

write this symbol just where you are before the tape head moves

So, still

Q = set of states

Σ = alphabet of input



$$\Gamma = \Sigma \cup \{\sqcup\} \cup \{\text{finite \# of symbols}\}$$

Γ: tape symbol alphabet

$\sqcup$: blank symbol, not part of Σ

q_0 has — initial state

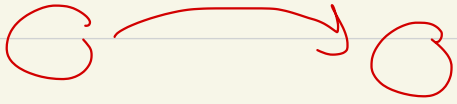
— accept state

— reject state

not equal

Diagram

$a \mapsto b, L$

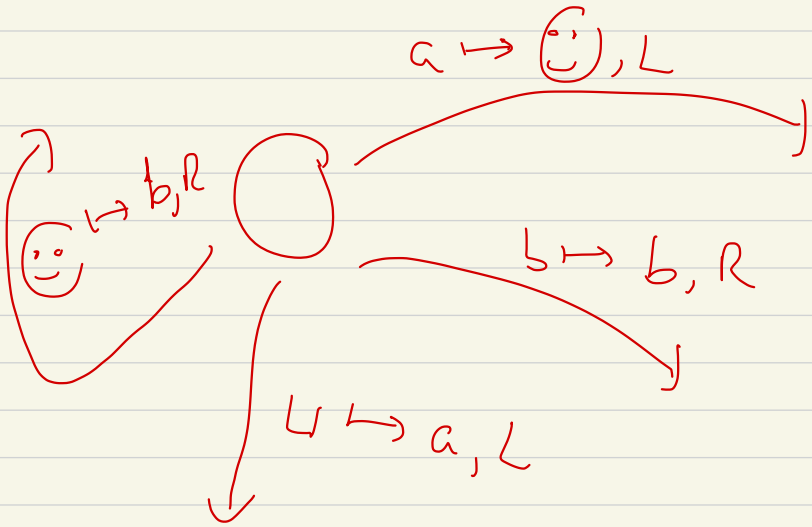


← what you have to write

e.g. $\Sigma = \{a, b\}$

$\Gamma = \{a, b, \sqcup, \text{☺}\}$

at each state:

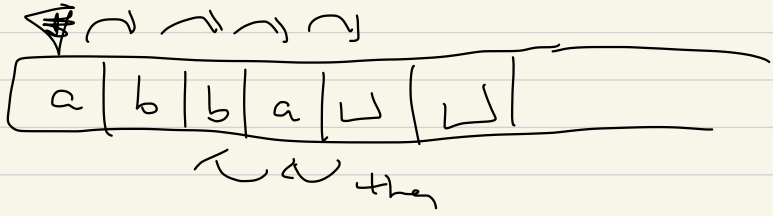


we have shortcuts...

Last time: $\Sigma = \{a, b\}$

$$C_2 = \left\{ w \in \Sigma^* \;\middle|\; \begin{array}{l} 2^{\text{nd}} \text{ to last} \\ \text{symbol of } w \\ \text{is an "a"} \end{array} \right\}$$

High-level description of TM:



① Move to the right until we see a ␣, then

② Move two steps to the left, and accept/reject according to whether or not we see an "a"

"Implementation-Level"

Start in q_0 (why not?)

keep moving to R when we see an a, b

(keeping in mind that we need to be careful when input, $w \in \Sigma^*$ is of length 0 or 1)

Once we've seen a $\sqcup$, move to $q_{\text{we've seen a blank}}$, then move the left 2 steps, one new state per

step, and then move to q_{acc}

or q_{rej} accordingly

=

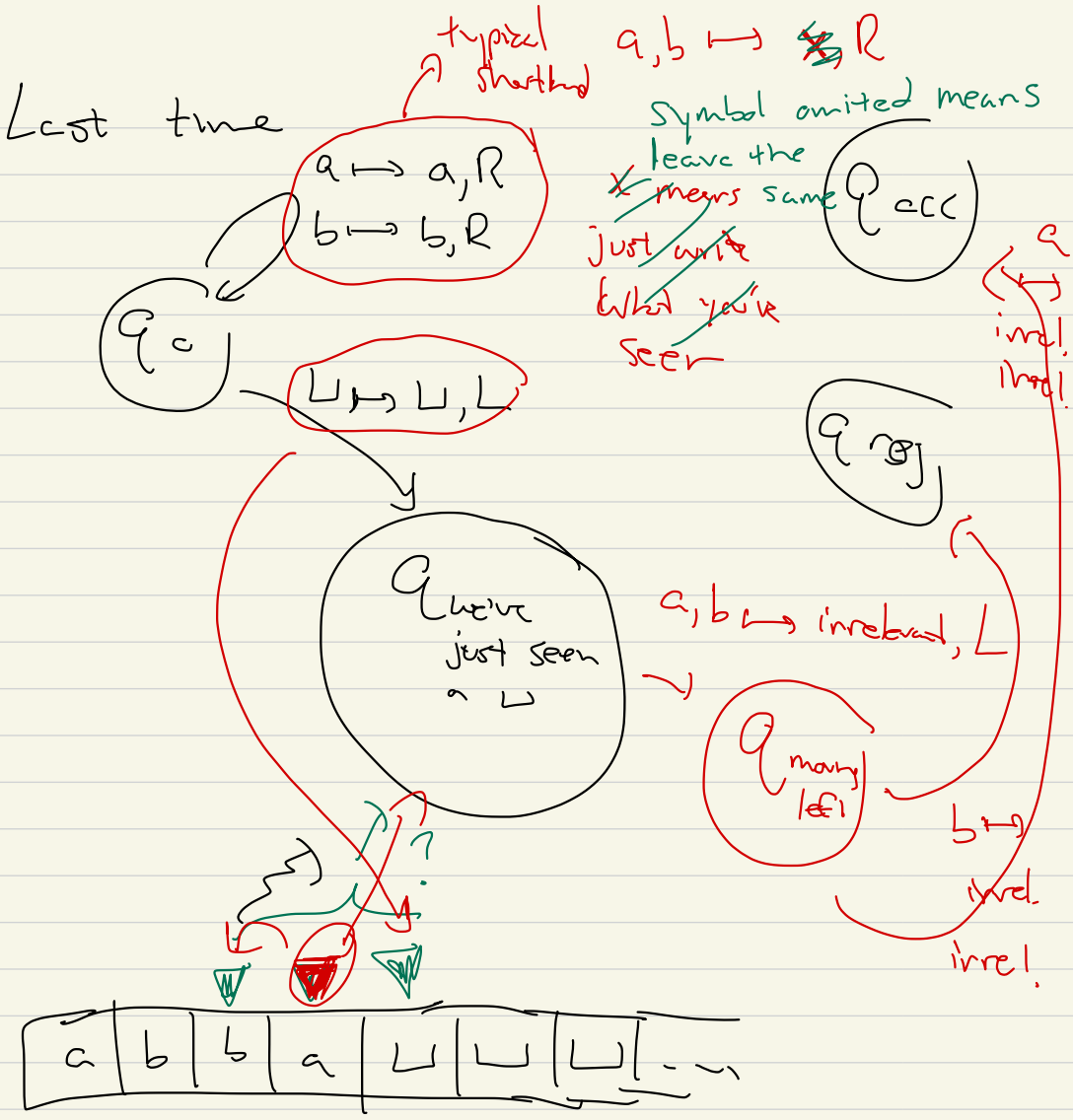
formal description: list $Q, \Sigma,$

$\Gamma, q_0, q_{acc}, q_{rej}$

specify δ by
- diagram
- table
- write out each value

(with some shortcuts)

Remark: After class: [skip] omits symbol if unchanged

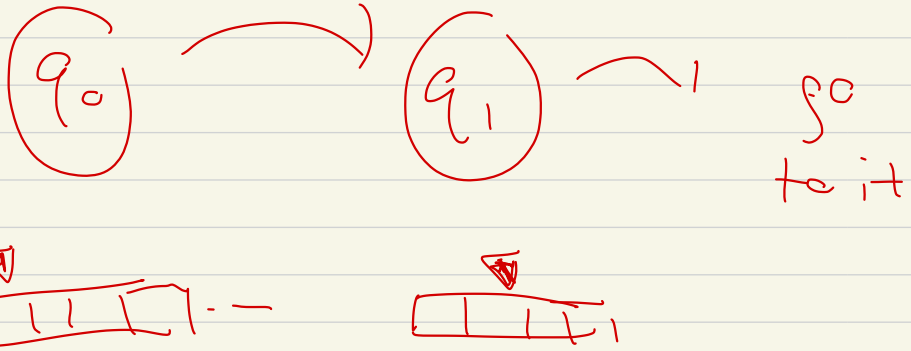


now

On input $w = \varepsilon$: type ␣ ␣ ␣ ...

length 0

(1) Could replace q_0 with



$w = \varepsilon$, tape $\sqcup\sqcup\cdots$

$w = a$ or b tape $a\sqcup\sqcup\cdots$
$b\sqcup\sqcup\cdots$

(2) Could add to Γ?

Now: $\{0^n 1^n\}$ or PALINDROME

Break 10:20 – 10:25

[Sip] says if you move L on cell #1, you just stay

Now

$\{0^n 1^n\}$ (1)

or

$PALINDROME_{0,1}$ (2)

$= \{ w \in \{0,1\}^* \mid w = w^{rev} \}$

$PALINDROME_{a,b} = \{ \varepsilon, a, b, aa, bb, \ldots abba \ldots \}$ (3)

(not ab, ba)

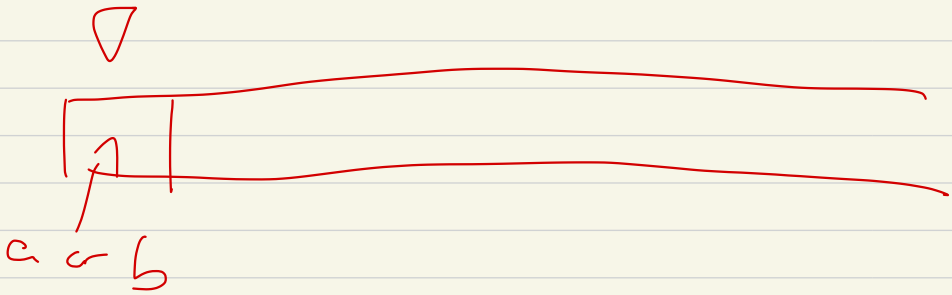
(2 or 3) !

$\Sigma = \{a, b\}$

$$PALINDROME = \{ w \in \Sigma^* \mid w^{rev} = w \}$$

$$= \left\{ \sigma_1 \sigma_2 \cdots \sigma_n \;\middle|\; \sigma_i \in \{a, b\} \right.$$
$$\left. \sigma_n \sigma_{n-1} \cdots \sigma_1 = \sigma_1 \cdots \sigma_n \right\}$$

n = length of input



branch to two branches,

one : $q_{\text{saw a}}$ a

$q_{\text{saw b}}$ b

move R until see ␣,

move L, compare what we

see

$$\Gamma = \{ a, b, \sqcup, a', b' \}$$

at q_0 we write

q_0 a b b a _ _ initial

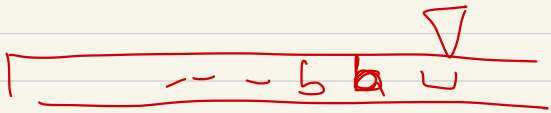
↓

a' b b a _ _ after 1 step

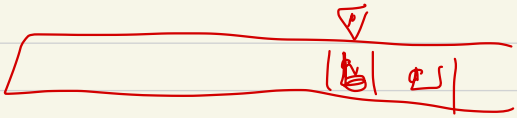
q_0 just saw a

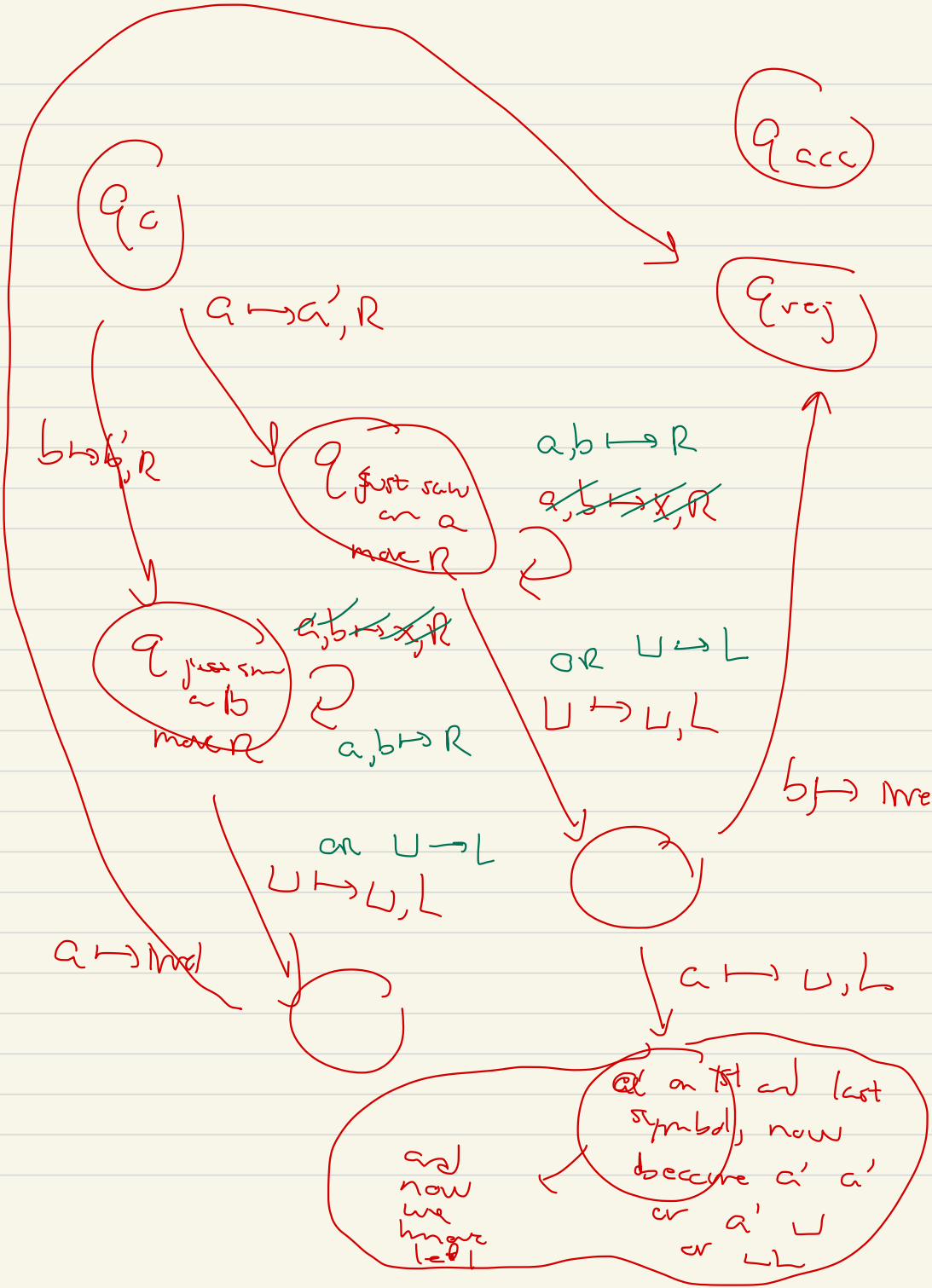
same

saw

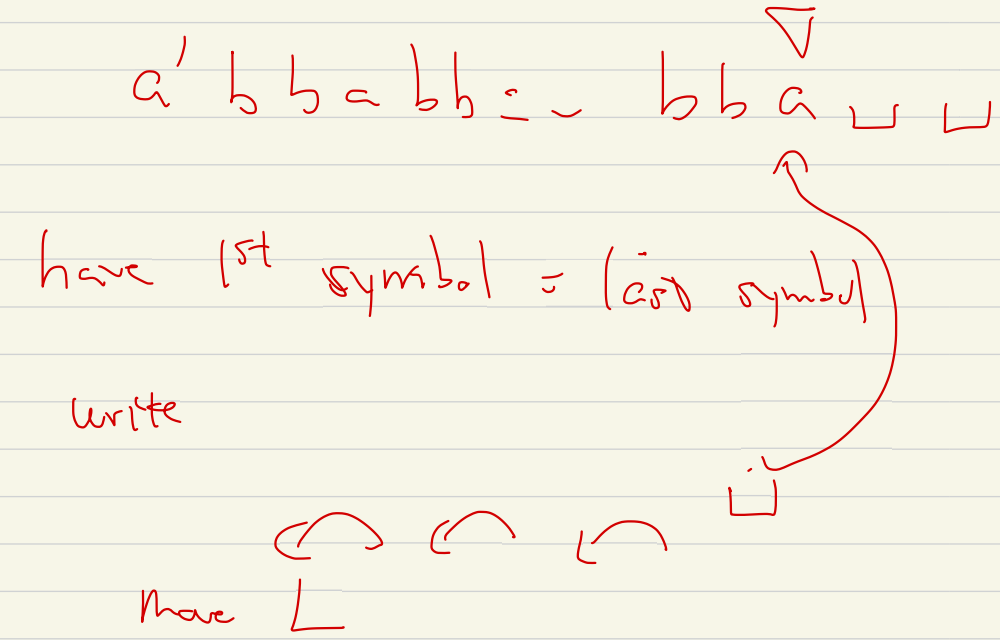


move




q_{acc}
q_c
q_{rej}
$a \mapsto a', R$
$b \mapsto b', R$
$q_{\text{just saw a}}$ move R
$a,b \mapsto R$
$a,b \mapsto x, R$
$q_{\text{just saw b}}$ move R
$a,b \mapsto x, R$
$a,b \mapsto R$
OR $\sqcup \mapsto L$
$\sqcup \mapsto \sqcup, L$
$b \mapsto$ rej
OR $\sqcup \mapsto L$
$\sqcup \mapsto \sqcup, L$
$a \mapsto$ rej
$a \mapsto \sqcup, L$
@ on 1st and last symbol, now become a' a' or a' $\sqcup$ or $\sqcup\sqcup$
and now we know left

Say:



This slide shown in parallel with last. Also, [S, p] writes $a \longrightarrow b, L$ instead of $a \longmapsto b, L$