CPSC 421/501 Oct 30

Multitape Turing machines

① Can be simulated by 1-tape T.m.'s

② Can be used to build a T.m. that solves

- MULTIPLICATION
- multiplication
- Acceptance$_{DFA}$
- Acceptance$_{Duck}$
- Acceptance$_{T.m.}$ universal Turing machine
- Acceptance$_{Python}$

Midterm on Friday

- In IRC 6, not IRC 4
- Wait outside until called in
- You will likely be seated in order of Last Name

- My office hours tomorrow: Room ICCS 146

Multiplication!

$$\begin{array}{r} 12345678 \\ \times\ 98765432 \\ \hline \end{array}$$

(one type)

one line

running total

+

one line

many lines

- combine to a finite number
- adding no problem
- keep a running total

Another method:

exponential time (that OK

before Ch 7. [Sip])

12345678

$\boxed{98765432}$

98765431 — decrement by 1

add

12345678 — to

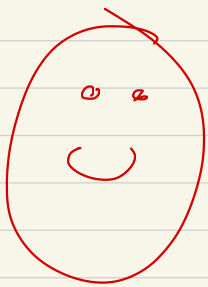
$\boxed{12345678}$ running total

Decrement again

98765431

98765430

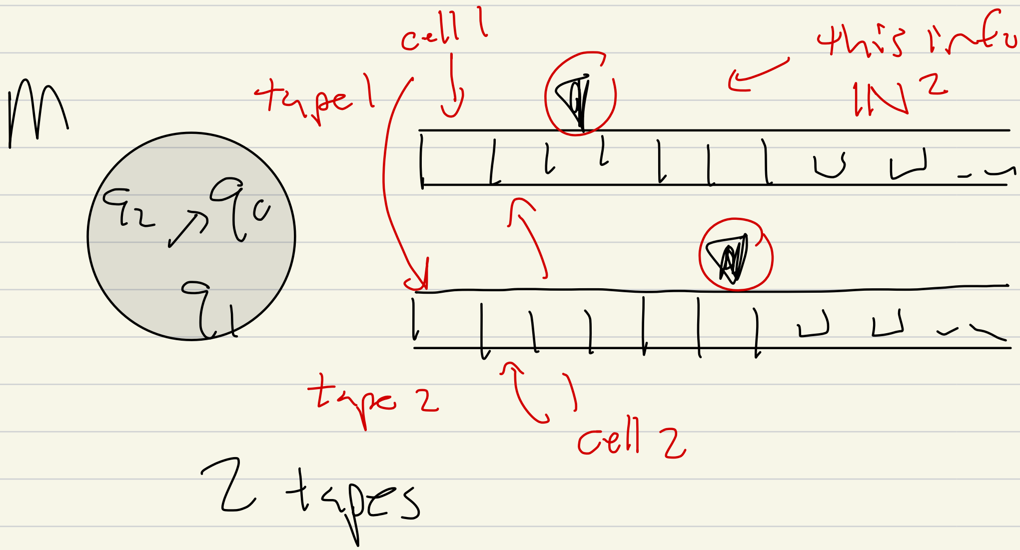
98765429



You

your boss

Remark: If L can be decided by a multi-tape machine, then it can be decided by a 1-tape machine



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

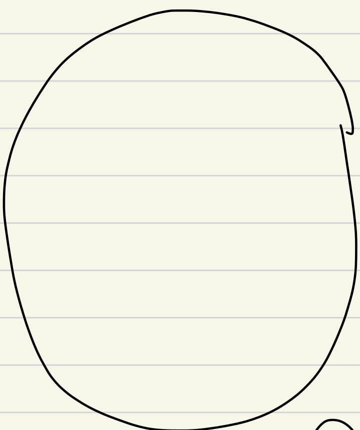
2 tape!

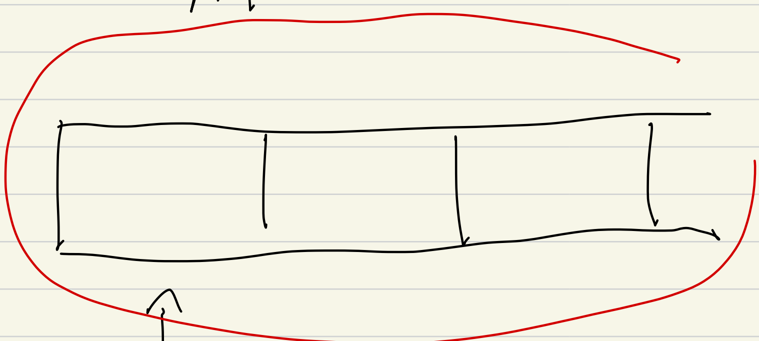
$$\delta : Q \times \Gamma^2 \to Q \times \Gamma^2 \times \{L, R, S\}^2$$

2 symbols on tape alphabet

How to simulate with 1-tape machine

M'





state set Q' can be much larger than Q

tape symbol

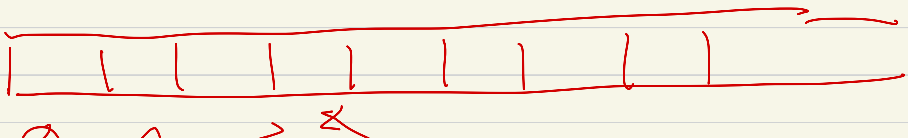
Γ' can be much larger

Rem: We want to know where ⊔ ⊔ ⊔ ... begins on both tapes

$\Rightarrow$

Possible?

symbols remember Γ, {yes, no}, is tape head there

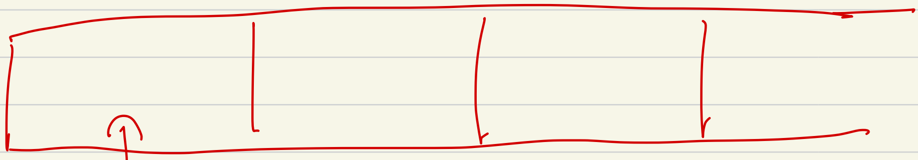


type 1 cell 1 — type 2 cell 1 — type 1 cell 2 — ...

is tape head 1 on cell 1? of tape 1

- - -
- - 2
- -
- - $\not{1}$
of tape 2

Possible?



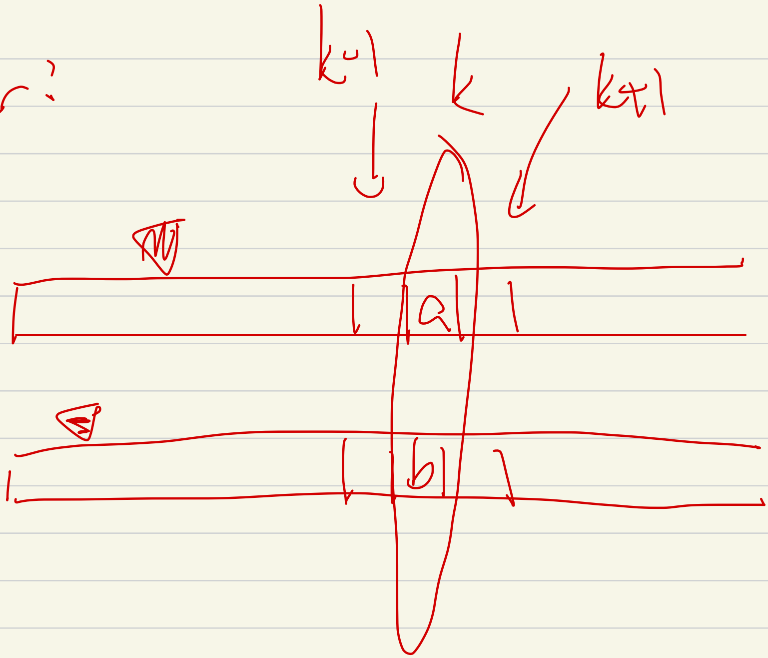
cell 1 of M's tape?

same with cell 2

What is on cell 1 of tape 1
" " on " 2 " " 2

is tape head 1 over cell 1
" " " 2 " " 1

Easier?



say that at cell k, we know that the two tape heads are not over cells k-1, k, k+1

To be continued ...

Sample Midterm problems ...

① The set of all regular languages over $\{a, b\}$ is countable

$$DFA = (Q, \Sigma, \delta, q_0, F)$$

can assume $Q = \{1, \dots, n_Q\}$

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

$\delta : Q \times \Sigma \longrightarrow Q$

$\{1, \ldots, n_Q\} \times \{a, b\} \longrightarrow \{1, \ldots, n_Q\}$

$q_0 \in \{1, \ldots, n_Q\}$

could assume $q_0 = 1$

So $n_Q = 1$, # DFAs as above

is finite.

If $n_Q = 2$, #DFA's is finite ...

$$\text{countable} = \begin{cases} \text{finite} \\ \text{countably infinite} \end{cases}$$

$\Updownarrow$

in bijection with

$\mathbb{N} = \{1, 2, \ldots\}$

Σ^* for alphabet Σ

$\{\text{All languages over } \Sigma\} = \text{Power}(\Sigma^*)$

Σ^* is countably infinite

$\Rightarrow$ Power(Σ^*) is uncountable

(Cantor's thm)

Alt proof: Regular language can be described by a regular expression:

over $\{a,b\}$

$$(a \cup bb)^* \circ (ba) \cup (abc)^*$$

Symbols: $a, b, (,), \cup, \circ, *, \varepsilon, \emptyset$

Map

$$\text{Reg Expressions} \overset{\text{Injection}}{\hookrightarrow} \underbrace{\{a, b, (,), \cup, \circ, *, \varepsilon, \emptyset\}^*}_{\text{countably infinite}}$$

T/F!

If L_1, L_2 unrecog, is $L_1 \cup L_2$ unrecog } F

False. —
$$\begin{cases} L_1 = \text{Accept}_{TM}^{comp} \\ L_2 = \text{Rejection}_{TM}^{comp} \end{cases}$$

$L_1 \cup L_2 =$ Everything