CPSC 421/501, Oct 20, 2020

§ 3.2 — Multitape vs. Single Tape

Skip for now

- Non-deterministic (more important for NP) (Chapter 7)

  $\delta : Q \times \Gamma \to \text{Power}(Q \times \Gamma \times \{L, R\})$

§ 3.3 — Descriptions of
- Graphs
- Boolean formulas
- etc.

---

Chapter 4: Goal: Find problems that are undecidable and unrecognizable

§ 4.1 : Decidable Problems

versus Recognizable Problems (examples)

§ 4.2 : Undecidable Problems

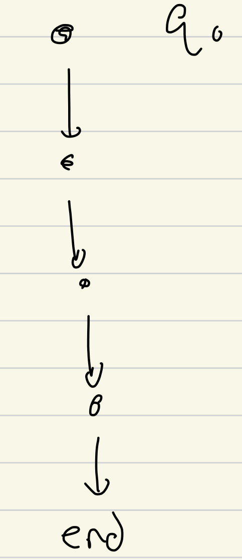
- Universal Turing Machines can recognize

  $A_{TM}$, $HALT_{TM}$

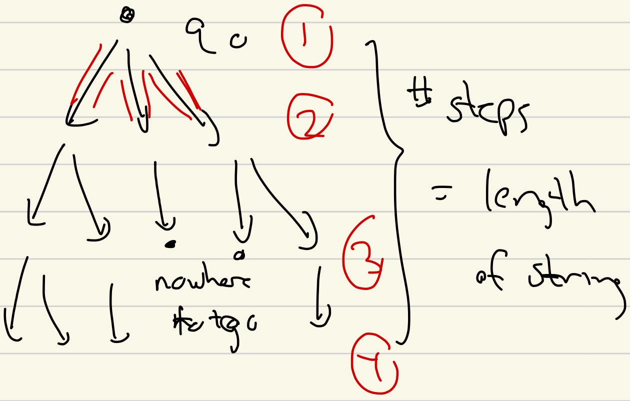
- But $A_{TM}$, $HALT_{TM}$ are undecidable.

For now we'll skip of non-determinism,
but the idea is like NFA

DFA



NFA:
$\delta : Q \times \Sigma \to \text{Power}(Q)$



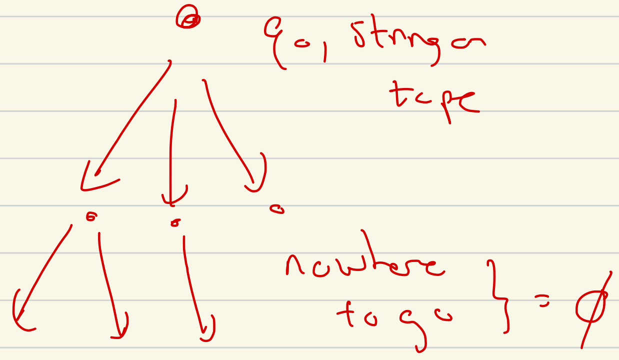
in § 3.2
you could simulate deterministically by a "breadth first search"

Nondeterministic TM

$$\delta : Q \times \Gamma \to \underbrace{\text{Power}(Q \times \Gamma \times \{L, R\})}$$

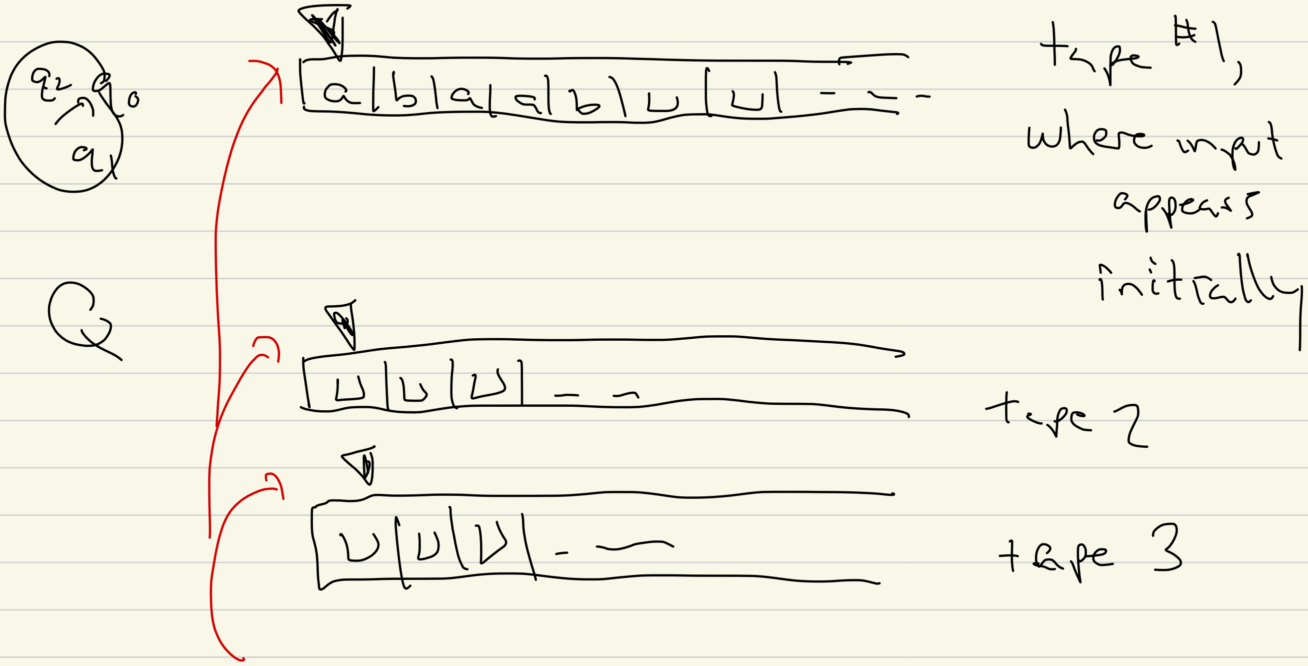
tape symbol

$\emptyset$ is a possible value of $\delta$



Nondeterminism important for NP (versus P)

Last time: Multitape machines.



$q_2$
$q_0$
$q_1$
Q
a | b | a | a | b | ⊔ | ⊔ | - - -
tape #1, where input appears initially
⊔ | ⊔ | ⊔ | - -
tape 2
⊔ | ⊔ | ⊔ | - -
tape 3

$$\delta : Q \times \Gamma^3 \longrightarrow Q \times \Gamma^3 \times \{L, R, S\}^3$$

stay

e.g. Recognize $\{0^n 1^n \mid n = 1, 2, \dots\}$

takes roughly quadratic time on 1-tape

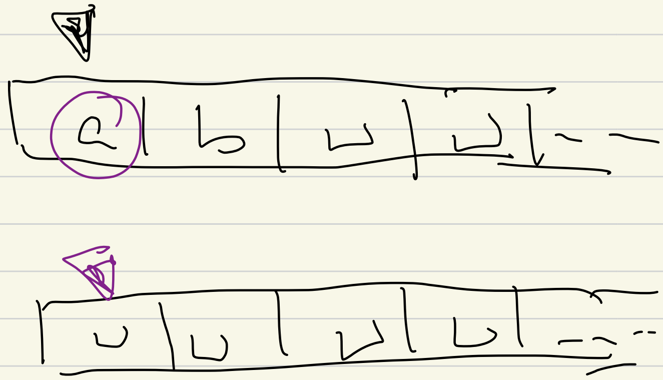
" linear time on 2-tape machine

Earlier: ∃ "High-level TM description"

a bit → Implementation- or Mid-level

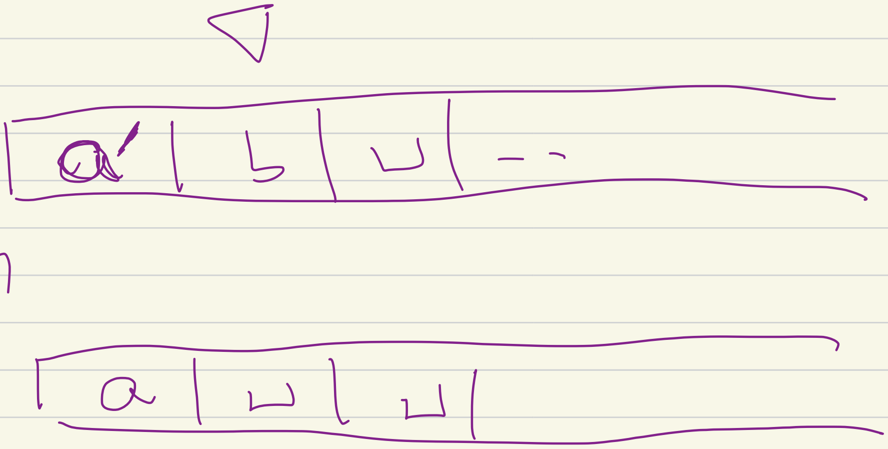
often ☹ - Describe δ entirely

Rem:



To copy tape 1 to tape 2

First step



a' new symbol to say "a was here"

and this is the left edge of tape 1

Rem: $\Gamma$ = symbols written to each tape cell is an <u>alphabet</u>, so a finite set

$$\Gamma = \Sigma \cup \{\sqcup\} \cup \{0,1,2,\ldots,15\}$$

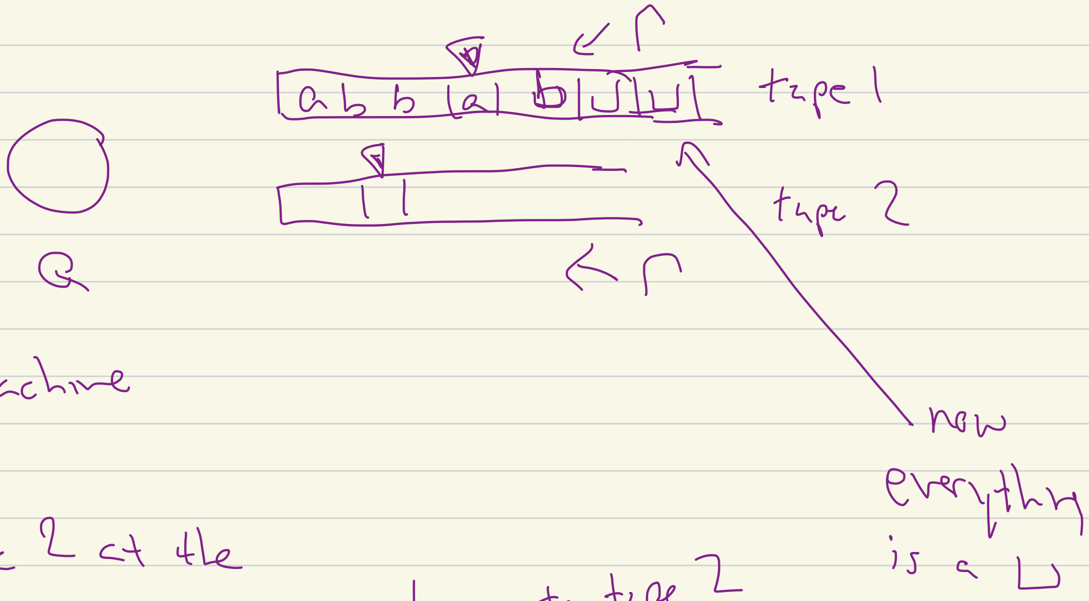
e.g. $\text{TIMES} = \{ a\#b\#c \mid a,b,c \in \{0,1\}^* \text{ s.t. } \text{binary}(a) \times \text{binary}(b) = \text{binary}(c) \}$

convenient to have a number of tapes

Theorem: If there is a multitape TM reccognizes L, then there is a 1-tape machine that reccognizes L.

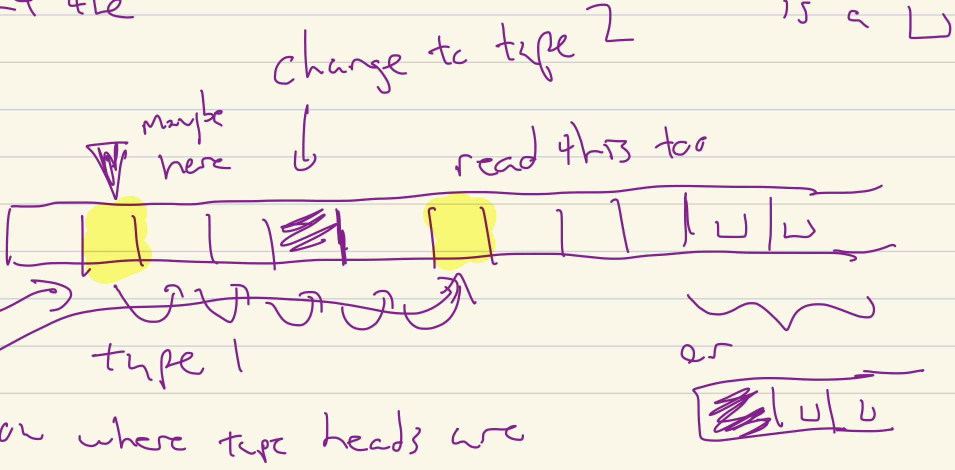
Ans: Yes (if you don't mind taking more time).

Idea:

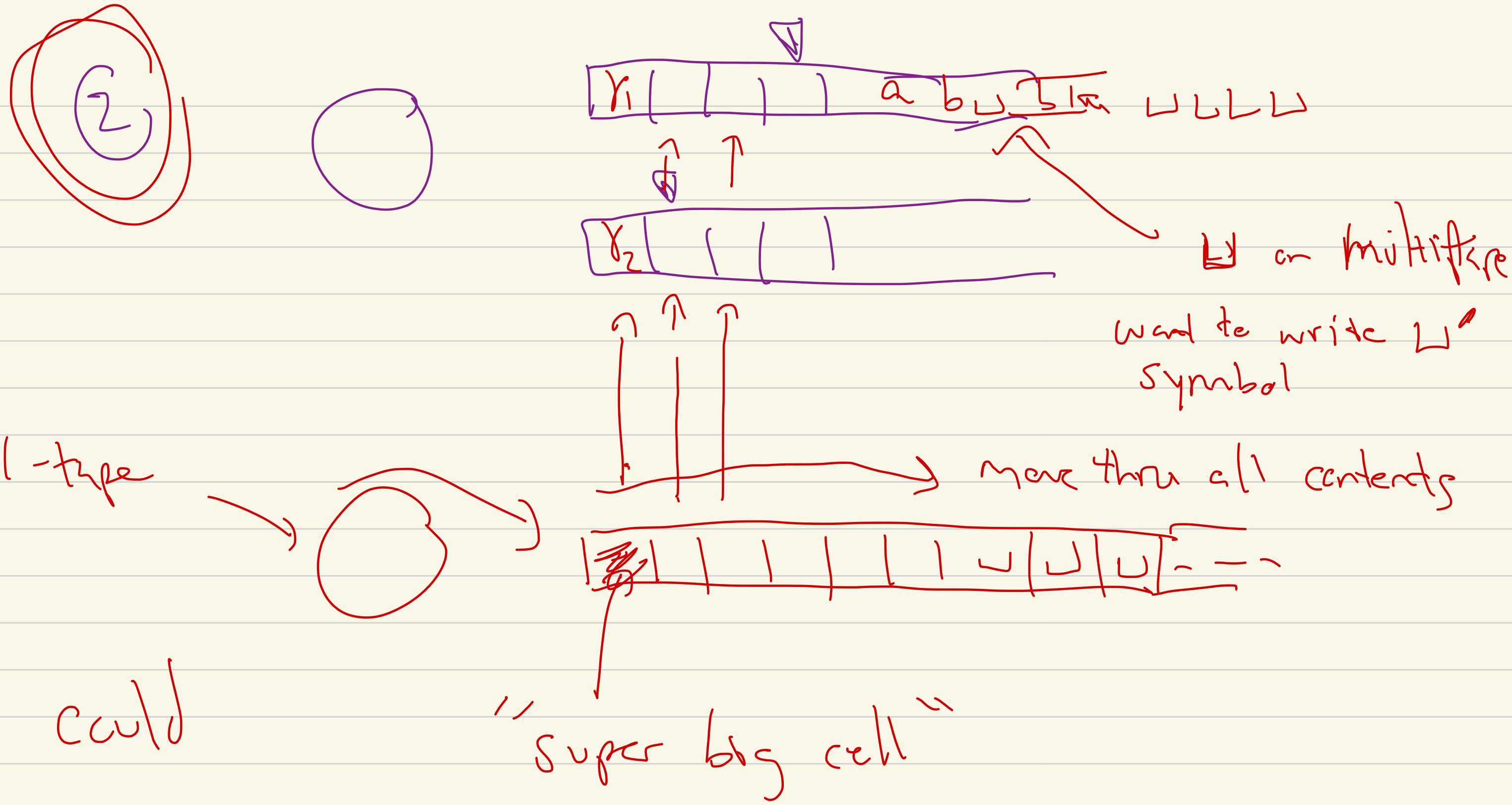


simulate this on a 1-tape machine

① put tape 2 at the end of tape



extra symbols or cells to tell you where tape heads are



telling you (1) what is $\gamma_1$ on tape 1 at that position

(2) is tape head 1 at cell 1?

(3) what is $\gamma_2$ - - - tape 2?

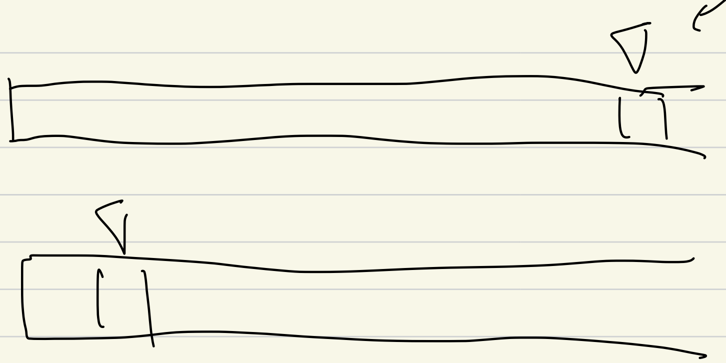
$\Gamma \times \Gamma \times \{$yes: is tape head 1 here, no$\} \times \{$yes, no$\}$

first tape, second tape

Turns out: any 2-tape machine algorithm that takes time = time(w) w = input, you may need $(\text{time}(w))^2$ on a 1-tape

but: 3-tape alg, still $(\text{time}(w))^2$ on a 1-tape

Real issue



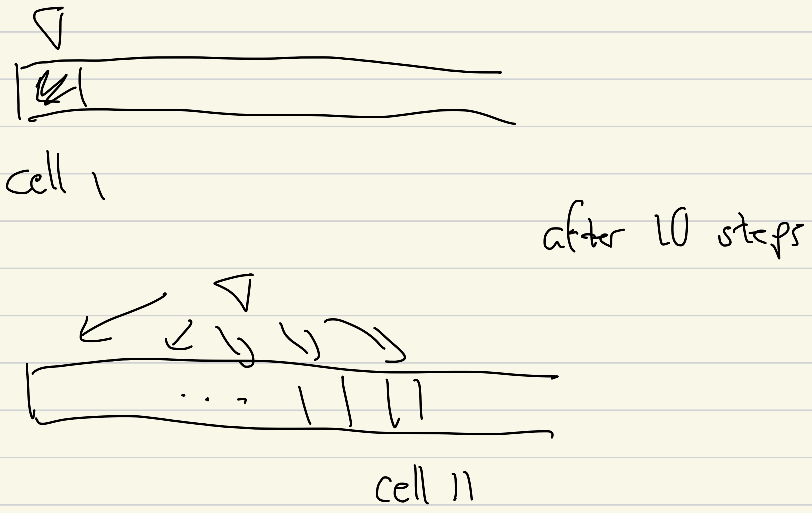
farthest ∇ can move is # steps taken so far

need to "run thru the whole tape contents"

=

**time** a TM takes on input w $\overset{def}{=}$ # steps until acc or rej

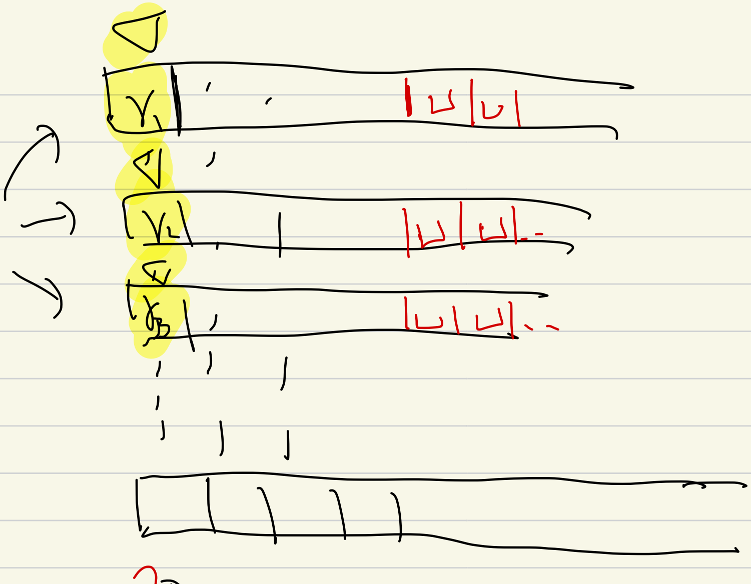
**space** a TM takes on input w $\overset{def}{=}$ farthest cell position (to the right) that a ~~tape~~ head reaches



**space ≤ time + 1**

So 3-tape

initially at cell 1,



super tape cell

1-tape: cell 1 tells you what is going on at cell 1 of tapes 1,2,3

if $\Gamma$ symbols in multitape

Super cell ⟷ super big 1-tape $\Gamma'$ = $\Gamma \times \Gamma \times \Gamma \times$

tape heads ↓↓

$\Gamma'$ cell 1 initially: (first let of input, ⊔, ⊔, yes, yes, yes, ...)

Break 5 min! 10:24 → 10:29

Finish Ch 3!

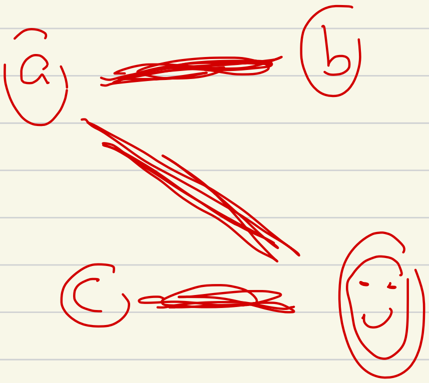
§3.3 : Only talk about "descriptions"

Say you have an algorithm to see

- if a graph is connected
- " " " " 3-colourable
- " " " " has a clique of size 5
- - - -~ - etc.

Technically graph $G = (V, E)$, here $V$ is a finite set, $E \subset$ unordered pairs of $V$

e.g.



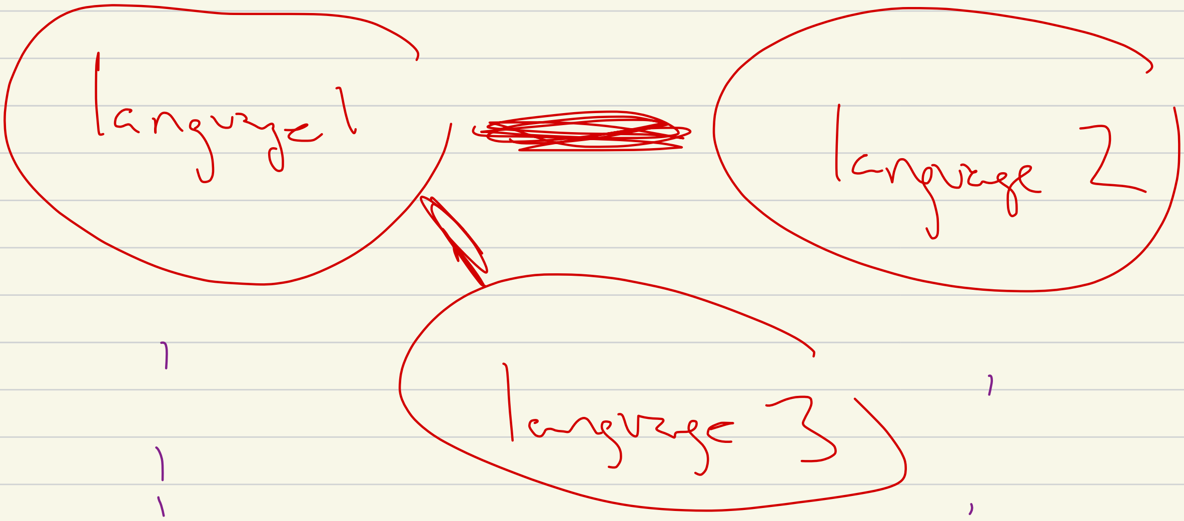
$V = \{a, b, c, \text{☺}\}$

$E = \{ \{a,b\}, \{a, \text{☺}\}, \{\text{☺}, c\} \}$

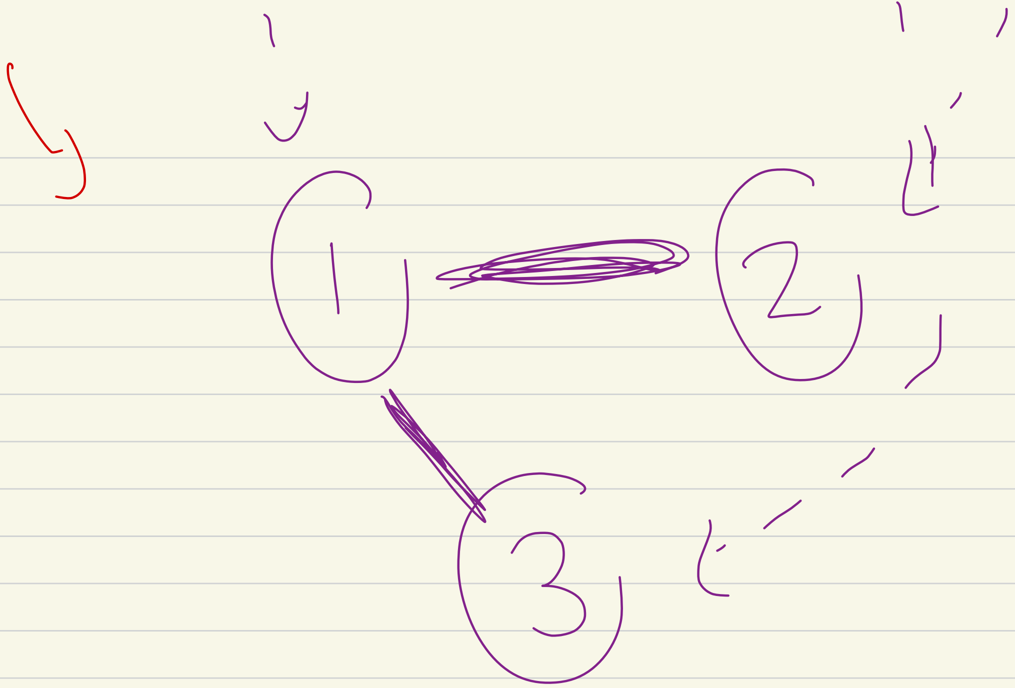
(directed graph: $E \subset V \times V = \{$ordered pairs of elements of $V\}$)

Question: {all graphs} countable?

Technically: $V$ could be a finite set of $\mathbb{R}$, or of some uncountable set



But, up to renaming, the vertex set



Technically!

- {set of graphs} uncountable
- {set of graphs s.t. $V = \{1, \dots, n\}$ for some $n$}

"Standard graph"

is countable

When you tell a computer about a graph,

You can say

$$\left\{ 26 \text{ vertices}; \text{ edges: } \{1,2\}, \{5,8\}, \ldots \{3,12\} \right\}$$

as a string

$$26 \# \underbrace{1 \# 2}_{\text{one edge}} \# \underbrace{5 \# 8}_{\text{another edge}} \ldots \# \underbrace{3 \# 12}$$

$$\in \{0, 1, \ldots, 9, \#\}^*$$

So we speak of

$$\langle G \rangle$$ as describing a standard graph

"description of G"

each should be a finite string

Similarly: Boolean formula:

$$f = (x_1 \wedge x_2) \vee \neg x_{35}$$

$\langle f \rangle$ = $( x 1 \wedge x 2 ) \vee \neg x 3 5$

"standardized Boolean formula"

$\in \{ (, ), x, \wedge, \vee, 0, \ldots, 9 \}$

---

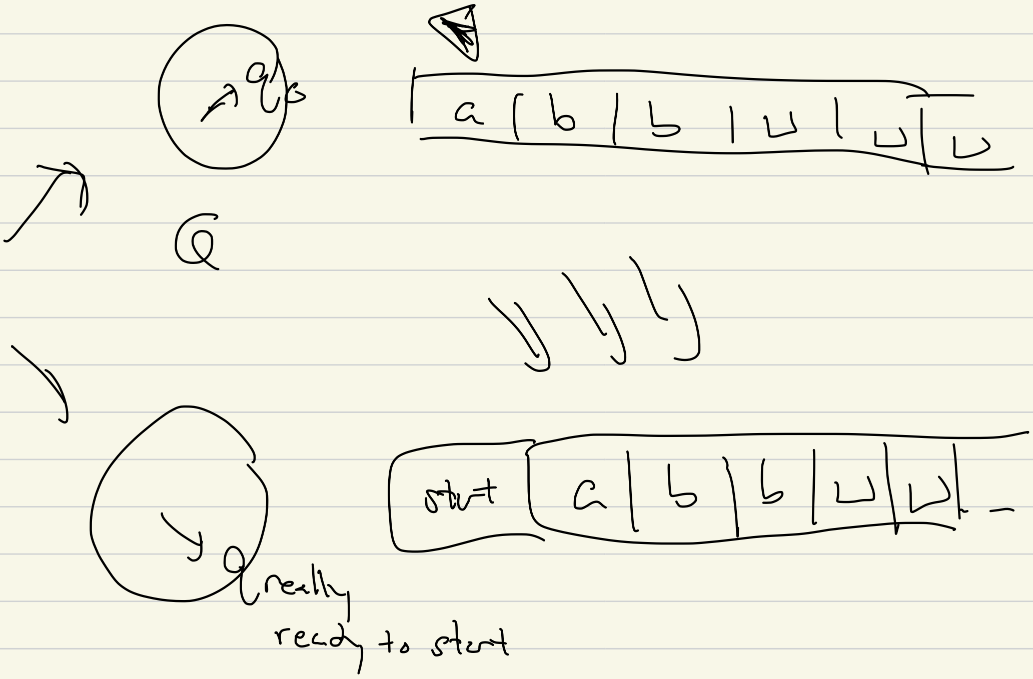
① Is the set of TM countable

② Can you "standardize TM" to make them countable?

# After class

1 - tape

- for "high-level" it is OK to just say this is your first step
- for "formal description" you have to give δ



q
a | b | b | ⊔ | ⊔ | ⊔ | ⊔
Q
q
start | a | b | b | ⊔ | ⊔
really ready to start