CPSC 421/501 Nov 8, 2024

Ch 7 : P vs. NP

Ch 8 : Space Complexity

+ Ch 9

how not to solve P vs NP

Ch 9 : How to solve P vs NP.

. . .

(even an easier problem . .)

P = Poly Time

= Poly Time for Turing machines

Polynomial:

If $f, g : \mathbb{N} \to \mathbb{R}$,

$g : \mathbb{N} \to \mathbb{R}_{>0}$

then

$$f = O(g)$$

$$f(n) = O(g(n))$$

if there is a C and n_0

s.t.

$$|f(n)| \le g(n)\, C$$

for all $n \geq n_0$

O = big oh notation

Also

$$f(n) = o(g(n))$$

$$\lim_{n \to \infty} \frac{f(n)}{g(n)} = 0$$

e.g.

$$f(n) = n^2 - 3n \begin{cases} n^2 + O(n) \\ O(n^2) \\ o(n^3) \end{cases}$$

We say that a Turing machine

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

runs in time $f(n)$,

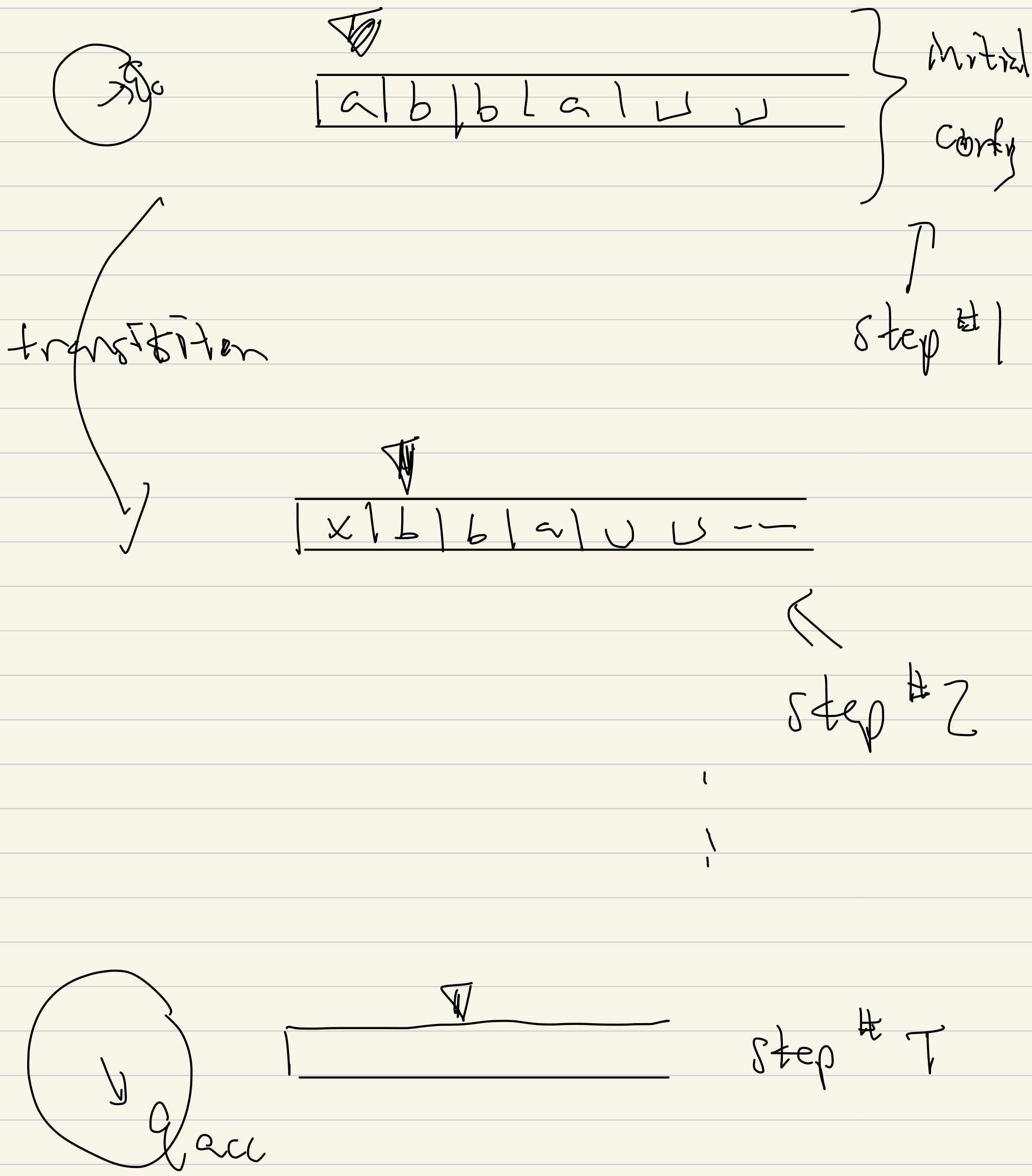
where $f(n) : \mathbb{N} \to \mathbb{R}$

$: \mathbb{Z}_{\geq 0} \to \mathbb{R}$

if on input $i \in \Sigma^*$, for any i,

M halts within time

$f(|i|)$



Turing machine takes time T on this

input.

input.	CPSC 320 intuitively time	time T.M.
So think sorting	$O(n \log n)$	$O(n^2)$
See if input is a palindrome	$O(n)$	2-tape T.M. $O(n)$

input abbbaaba

1-tape T.M. at least $c\,n^2$ time

difficult theorem

Claim: In most models of computation (Python, C++, Javascript) classical type, if L can be decided in poly time, $O(n^k)$ for some $k \in \mathbb{N}$, then L is decidable in time $O(n^{k'})$ for some $k' \in \mathbb{N}$.

$P = \{ L$ over a finite alphabet, Σ, that can be decided in time $\leq C n^k$ for some

C, k, i.e. in time

$O(n^k)$ some k }

CPSC 320 $\Rightarrow$ many L

that can be decided in time

$O(n^k)$ $k = 1, 2, 3.$

e.g.,

$L = \{ \langle G \rangle \mid$ G is a graph that is bipartite $\}$

$=$ BIPARTITE

Graph!

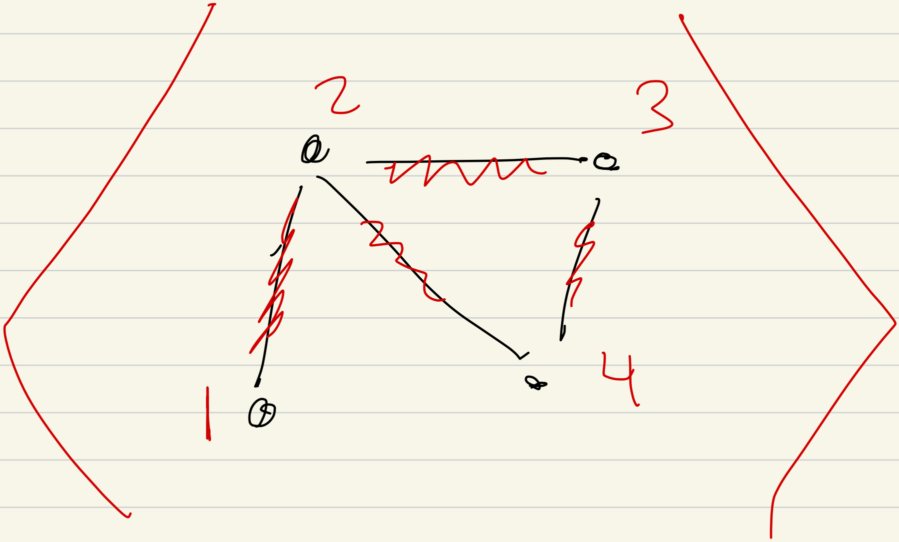
Graph: (V, E)

V is a set

E collection of unordered pairs of elements of V

$V = \{1, 2, 3, 4\}$

$E = \{ \{1,2\}, \{2,3\}, \{3,4\}, \{2,4\} \}$



$4 \# 1\#2 \# 2\#3 \# 3\#4$

$\# 2\#4$

↑

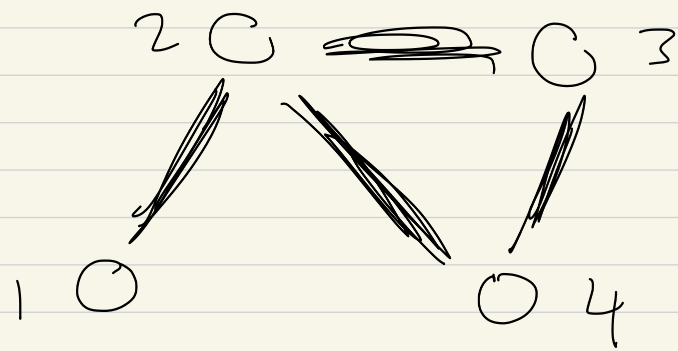
num of vertices

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

Graph G is 2-colourable

if $V \to \{red, blue\}$

s.t. each edge has different colours at its endpoints



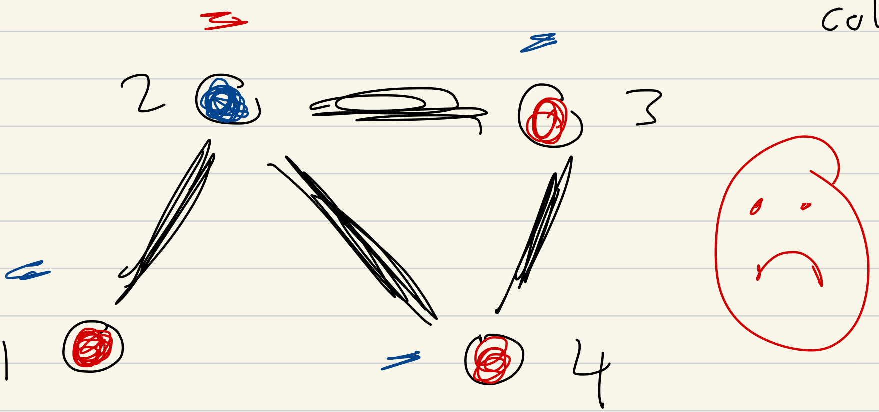
Alg 1: Try all possible

$V \to \{red, blue\}$

maps? $2^{|V|}$ finite

=

Alg 2! Start anywhere, pick a colour



Theorem! This alg find a "proper 2-colouring" if G is 2-colouring, and if alg doesn't work, then G is not 2-colourable

This implication idea really does prove: if the alg fails, then G is not 2-colorable.

=

Argue this is poly time

$$\underbrace{4 \# 1 \# 2 \# 2 \# 3 \# 3 \# 4 \# 2 \# 4}_{\langle G \rangle}$$

Alg is poly time in # vertices

3COLOR # 213 # 7 # 1 # 24

$\underbrace{}$

$|V|$

$$|\langle G \rangle| = O(\log(n)) + \begin{cases} \text{at least } 3m \\ O(m \log(n)) \end{cases}$$

$m = |E|$

$G = (V, E)$

2COLOUR

$$= \{ \langle G \rangle \mid G \text{ has a proper 2-colouring} \}$$

3COLOUR

$$= \{ \langle G \rangle \mid G \text{ has a proper 3 colouring} \}$$

$$G = (V, E)$$

$$V \longrightarrow \{\text{red}, \text{blue}, \text{green}\}$$

P vs. NP

$\iff$

Is

3COLOUR $\in$ P

is 3colour decidable

in poly time