

## Learning Goals

- Describe P and NP informally, and why these ideas are important
- Define NP
- Prove a problem is in NP [NP1]

## Announcements

CS Seminar on Friday, alum Corey go/meetCorey  
Reattempts → Office Hours!

Next exam:

- 75 min during class, Tues 3/17
- 1 side of 3"x5" notecard
- DC1, DC2, DP1, G1, NP1, DC3 ← Core
- You are responsible for knowing which targets you need

Middlebury College  
**Computer Science Seminar**

## Alumni Tech and Career Talk



### **Corey Scheinfeld** Software Engineer for Tradeweb

In this talk, Corey will discuss his experience transitioning from studying computer science at Middlebury to working as a software engineer at Tradeweb. He will describe the kinds of systems he works on, and the various pieces of the software development lifecycle that support high-throughput, market central financial platforms. The presentation will connect elements of his undergraduate coursework to their application in production environments, and highlight how a liberal arts background has been critical to his early career in technology. He will also discuss how AI has reshaped the industry, and what tools like Claude and Cursor have changed in the day to day life of a software engineer.

*Bio: Corey Scheinfeld grew up in New Rochelle, New York and attended Middlebury College from Fall, 2018 to Spring, 2022 and graduated with a degree in Computer Science. He has since worked as a software engineer at Tradeweb, an electronic trading marketplace for fixed income, derivatives, ETFs, and equity markets, where he also interned as a student at Middlebury. He works on their institutional technology platform, building scalable trading systems and infrastructure, particularly Node.js based microservices, operating in performance-sensitive environments. In this capacity, he leads a team of engineers on the development of various fixed income trading protocols. Outside of work, Corey enjoys skiing, hiking, and returning to Vermont as often as he can.*

**Friday, March 6, 2026 @ 12:20 p.m.**

**75 Shannon St 102**

***Pizza will be served***

go/meetCorey

# Warm-Ups

Why if  $x$  is a NO-instance, will  $M$  fail for all  $y$ ?  
<sup>how</sup>  
<sup>why</sup>

How to identify  $y$ ? (Your job!)

What if better than  $O(\text{poly}|x|)$  using not  
brute force? }  $\text{conxn}$

Does our NP proof translate into an alg? }

$$O(2^{|x|} \text{poly}|x|)$$

check all  $y$

$\Omega$  thing

# Types of Problems ? \$1 million

## Easy

(Polynomial time)  
 $O(n)$ ,  $O(n^2)$   
 $O(n^3)$

- Search
- Sort
- Multiplication
- Greedy Scheduling
- Closest Pts
- Matrix Mult.

Fast algorithm

Quantum Computers

## Puzzles

Crosswords ( $n \times n$ )

Sudoku ( $n \times n$ )

Delivery Rts

$n$  prime?

Factoring large #

MWS general  
graph  $\geq 1000$

Easy to check

Might be hard to  
solve

## Hard

Chess:  
What is next best  
move?

Can mathematically characterize Easy / Puzzle

P and NP

P (Polynomial Time)

Informal: A problem is in P if it can be solved in polynomial time

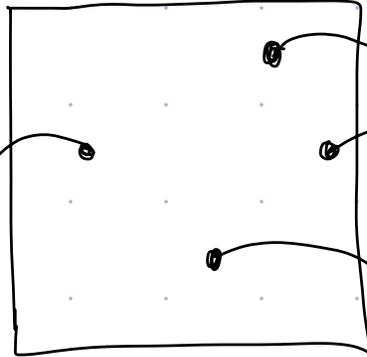
NP (Non-deterministic Polynomial Time)

Informal: A problem is in NP if a solution can be checked in polynomial time.

Polynomial Time ??

- Runs in time  $O(n^c)$  where  $n$  is the size of input and  $c$  is a constant.  $O(\text{poly}(n))$

All Problems



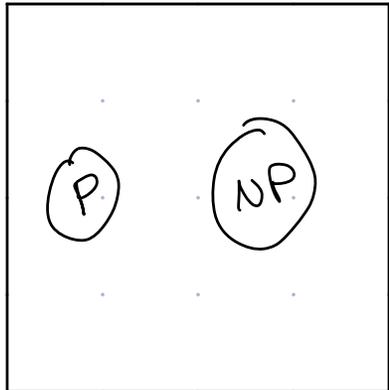
Chess next best move

Halting Problem

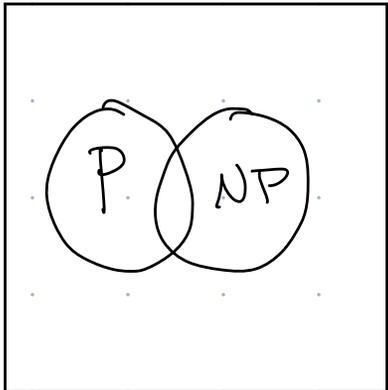
Sudoku

Closest Pts

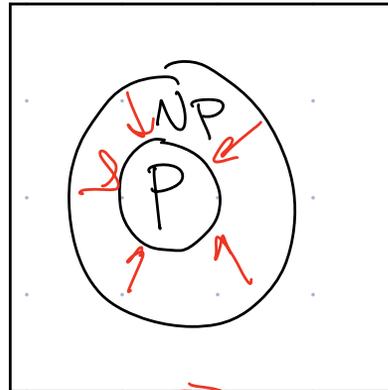
Which picture is correct?



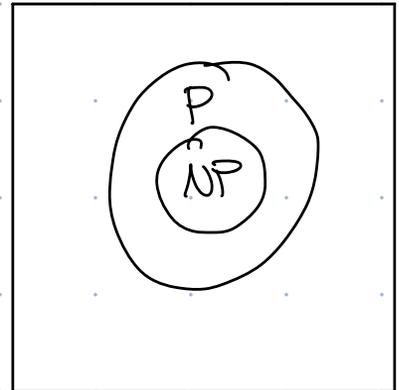
A X



B X



C



X D

NP problems are YES-NO:  $Q(x) = \begin{cases} \text{Yes} \\ \text{No} \end{cases}$

↑  
input

example of NP problem: 3SAT CNF formula

3SAT:  $x$  is a YES instance if it describes a Boolean formula that is an AND of ORs where each clause has at most 3 literals and there is an assignment of variables that makes  $x$  true

Instance:

$$x = (z_1 \vee z_2 \vee \neg z_3) \overset{\text{AND}}{\wedge} (\neg z_1 \vee \neg z_3 \vee z_4) \overset{\text{AND}}{\wedge} (z_2 \vee z_4) \wedge \dots$$

↑     ↑  
OR

$z_i, \neg z_i$

$z_1, z_2, \dots \iff$  variables

$\neg z_1, \neg z_2, \dots \iff$  literals

Otherwise  $x$  is a NO instance

Is 3SAT  $\in$  NP?

## Questions to Ask Yourself to Prove $Q \in NP$

① What info would convince me that  $x$  is a Yes for  $Q$ ?  
 $x$  is YES for 3SAT:

$$(z_1 = T, z_2 = F, z_3 = F \dots) \leftarrow y$$

② If given info in ①, how could I check if  $x$  is a Yes for  $Q$ , quickly?

3SAT: Plug  $y$  in and test each clause

★ You do not have to show how to find  $y$  ★

★ Only need to show how to verify  $y$  is solution to  $x$  ★

$$x = z_i \vee \neg z_i$$

$$y = z_i \leftarrow F/T$$

watch

# Proof that 3SAT ∈ NP

[NP1]

Let  $M(x, y)$

alg.  
instance  
potential

solution

be the alg that

- ① Check that  $x$  is a CNF formula with at most 3 literals per clause.
- ② Check that  $y$  is a assignment of T, F to each variable.
- ③ Check that with assignment  $y$ , every clause of  $x$  is true.

And output 1 if all checks pass, 0 otherwise.

- If  $x$  is a Yes instance, then a satisfying assignment  $y$  exists and  $M(x, y)$  will pass all checks. Otherwise if  $x$  is a NO instance, at least one check will fail for any  $y$ .
- If 3-SAT with  $n$  variables,  $m$  clauses then  $|x| = \Omega(m \log_2 n)$  so  $M(x, y)$  can be run in  $O(\text{poly}(|x|))$  time with brute force.

# Formal(ish) Definition of NP

A problem is in NP if

- YES-NO

- There is a polytime algorithm  $M$  s.t.

  - If  $x$  is a YES instance,  $\exists y$  s.t.  $M(x, y) = 1$

  - If  $x$  is a NO instance,  $\forall y, M(x, y) = 0$

"Decision Problem"

input  $x$

$O(\text{poly}(|x|))$

"verifier"

"witness"

0011

010011

1001100100

# Group Work [NP1]

Hamiltonian Path Problem:  $X$  is a YES instance iff  $X$  describes adjacency matrix of a graph  $G$  with vertices  $s, t$  such that there is a path from  $s$  to  $t$  that goes through each vertex exactly once.

$X =$

	s	u	v	t
s	0	1	0	0
u	1	0	1	1
v	0	1	0	1
t	0	1	1	0

Show:  $\uparrow$   
YES instance

$X =$

	s	u	v	t
s	0	1	1	1
u	1	0	0	0
v	1	0	0	1
t	1	0	1	0

$\uparrow$   
No instance

Prove: Hamiltonian Path  $\in$  NP

- Names, Pronouns
- Dining Hall Hack

- Describe  $M(x, y)$
- Lower bound  $|x|$  in terms of  $n$  (vertices),  $m$  (edges)  $\rightarrow$   $\text{poly}(|x|)$  runtime for  $M$
- Correctness ( $\exists y, \forall y$ )

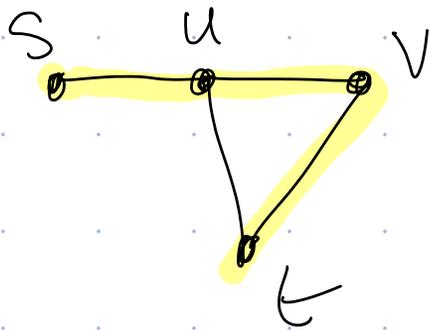
# Group Work

Hamiltonian Path Problem:  $X$  is a YES instance iff  $X$  describes adjacency matrix of a graph  $G$  with vertices  $s, t$  s.t. there is a path from  $s$  to  $t$  that goes through each vertex exactly once.

$$X = \begin{array}{c|cccc} & s & u & v & t \\ \hline s & 0 & 1 & 0 & 0 \\ u & 1 & 0 & 1 & 1 \\ v & 0 & 1 & 0 & 1 \\ t & 0 & 1 & 1 & 0 \end{array}$$

Show:

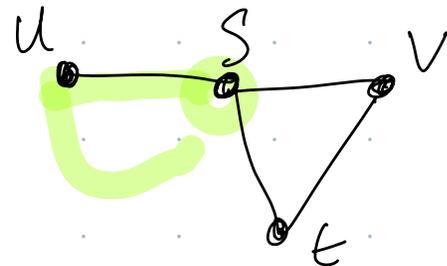
↑  
YES instance



$$y = (s, u, v, t)$$

$$X = \begin{array}{c|cccc} & s & u & v & t \\ \hline s & 0 & 1 & 1 & 1 \\ u & 1 & 0 & 0 & 0 \\ v & 1 & 0 & 0 & 1 \\ t & 1 & 0 & 1 & 0 \end{array}$$

↑  
No instance



$$X = 0100101101010110; s=1, t=4$$
$$|X| = n^2 + \log n$$

# Ham Path $\in$ NP

$$|x| = O(n^2)$$

$$O(n^2)$$



## • $M(x,y)$ check

① Check  $x$  is adj matrix with vertices  $s, t$ .

②  $y$  is a list of  $n$  vertices start at  $s$ , end at  $t$ .

③  $y$  has no repeats  $O(n)$   $O(1)$

④ For each consecutive pair  $(u,v) \in y$ ,  $(u,v)$  <sup>should be</sup> edge in  $G$   $O(n^3)$

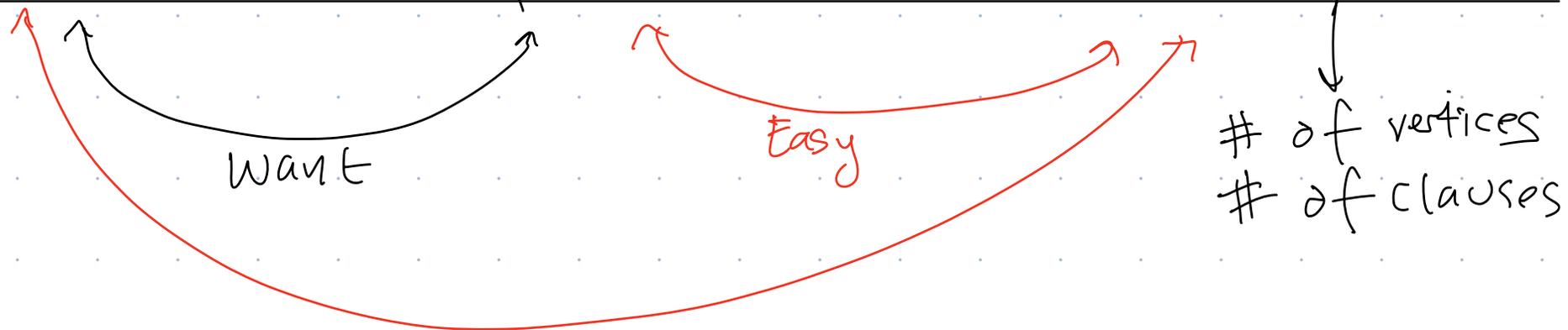
And output 1 if all checks pass, 0 otherwise.

• If  $x$  is a Yes instance, then a path  $y$  exists and  $M(x,y)$  will pass all checks. Otherwise if

•  $x$  is a NO instance, at least one check will fail for any  $y$ .

• If a yes instance has  $n$  vertices,  $m$  edges  $|x| = \Omega(n^2)$ , so  $M(x,y)$  can be run in  $O(\text{poly}(|x|))$  time with brute force

# Runtime vs Inputsize vs Natural Parameters



①  $|x| = \Omega(n^2)$

Easy

$\longleftrightarrow$  " $|x| \geq n^2$ "  $\rightarrow$   $|x|^{1/2} \geq n$   
 $n \leq |x|^{1/2}$

②  $T = O(n)$

$\longleftrightarrow$  " $T \leq n$ "

① + ②

$T \leq n \leq |x|^{1/2} \Rightarrow T \leq |x|^{1/2}$

$T = O(\text{poly}(|x|))$