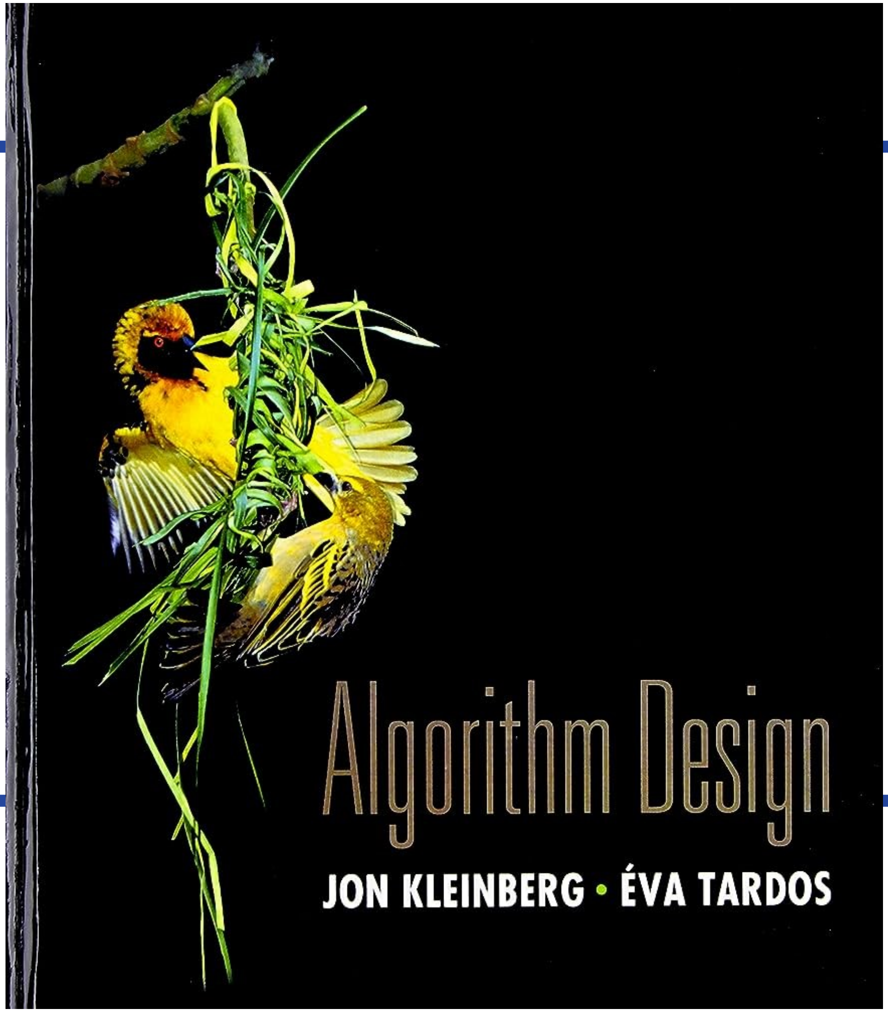
CS 310: Algorithms

Lecture 24

Instructor: Naveed Anwar Bhatti

Few Slides taken from Dr. Imdad's CS 510 course



Chapter 8:
NP and Computational Intractability

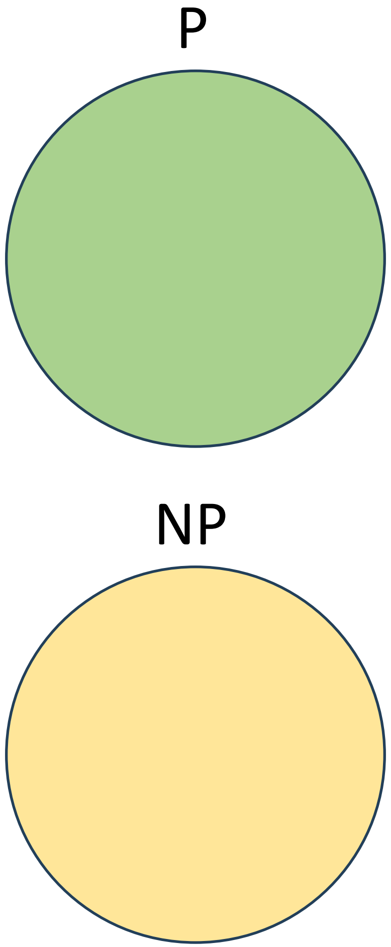
Section 8.3 :
Efficient Certification and the Definition of NP

Efficiently Solvable Problems

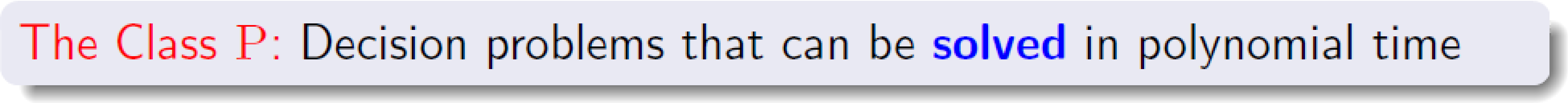
The Class P: Decision problems that can be **solved** in polynomial time

The Class NP: Decision problems that can be **verified** in polynomial time

NP stands for “Non-deterministic Polynomial Time”



Efficiently Solvable Problems

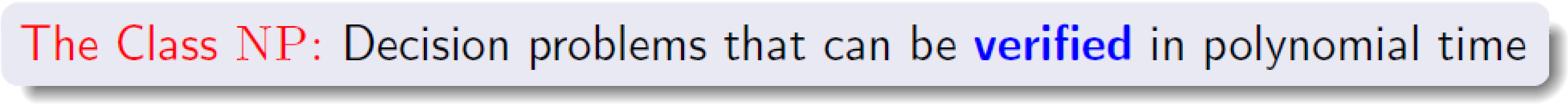


The Class P: Decision problems that can be **solved** in polynomial time

To prove problem **A** є **P,** we need to provide algorithm ***F*** and show:

- ***F*** runs in polynomial time
- On any instance of the problem **A**, ***F*** correctly output YES/NO

Efficiently Solvable Problems


The Class NP: Decision problems that can be **verified** in polynomial time

To prove problem **B** ϵ **NP,** we need to provide Certificate ***C*** and verifier **V**, and show:

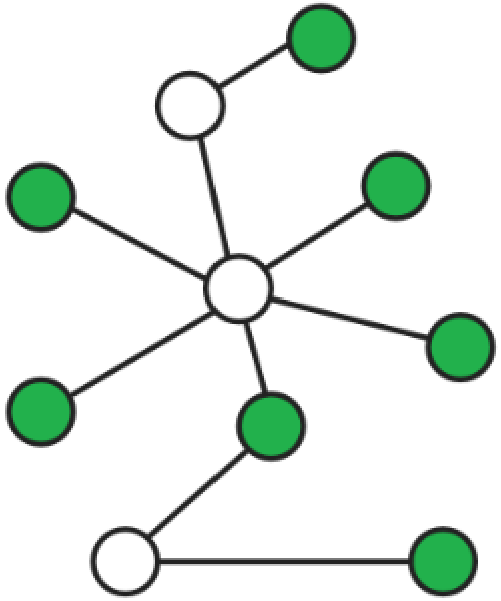
- **C** always has polynomial length
- **V** runs in polynomial time
- **V(b,C)** outputs **YES** *if and only if* **b** is a **YES** instance and **C** is a valid

Certificate and Verifier

Independent Set: G, K

Certificate:

An independent set **S**



Verifier:

- Check S ⊆ V(G)
- Check |S| ≥ K
- Set S is in fact independent set – meaning no edge exist between pair of vertices in set S

Certificate and Verifier

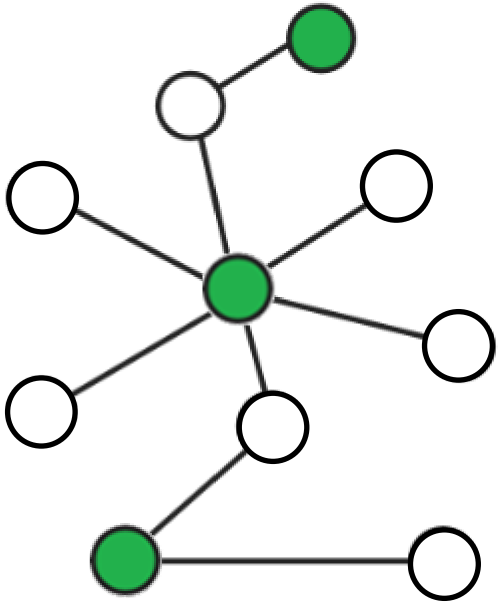
Vertex Cover: G, K

Certificate:

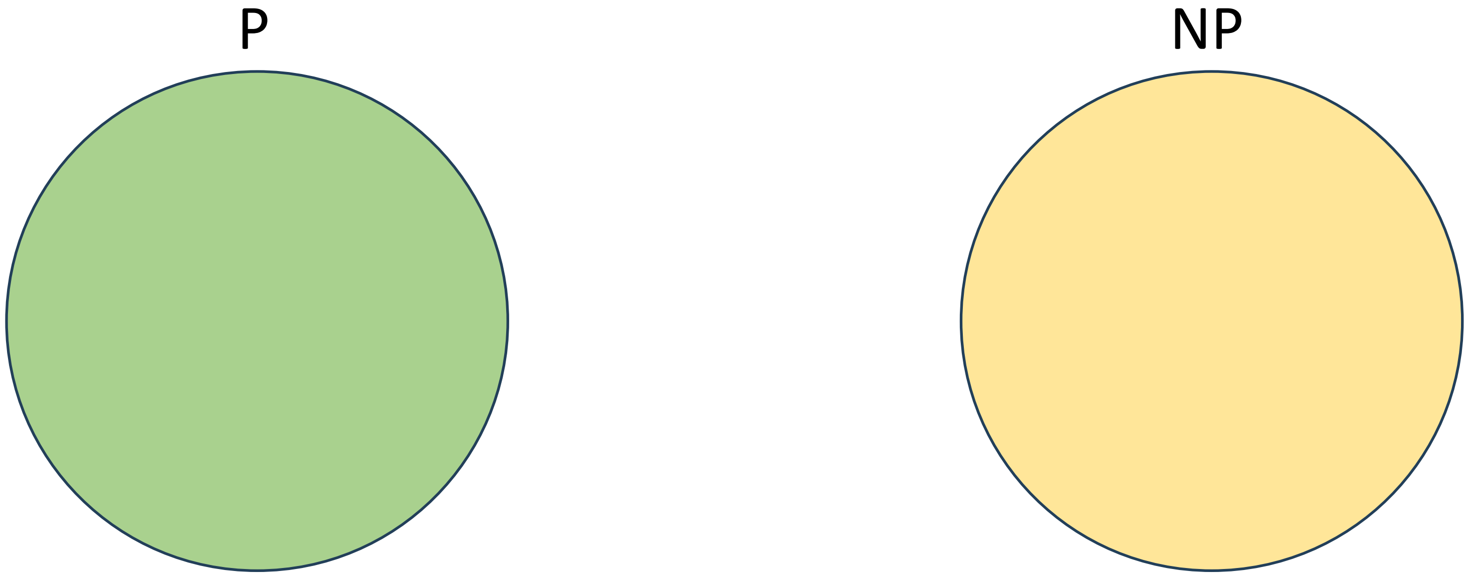
A vertex cover set **S**

Verifier:

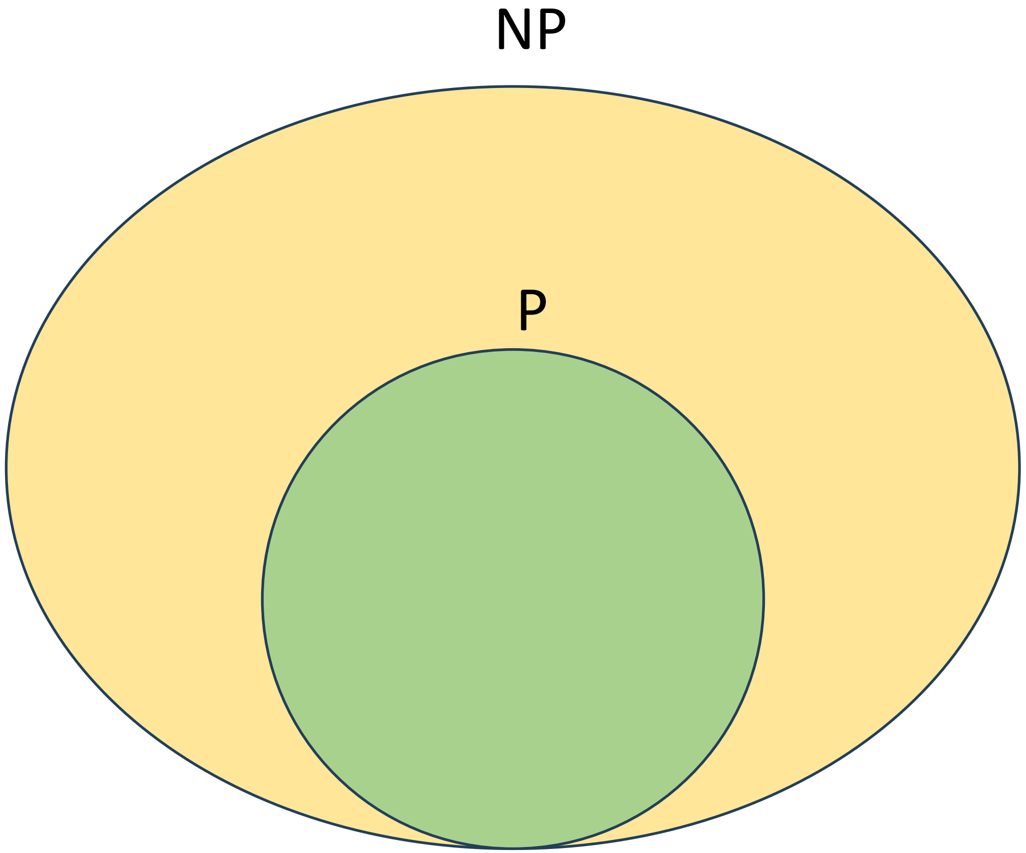
- Check S ⊆ V(G)
- Check |S| ≤ K
- For every edge in the graph at least one end is connected with vertex cover set



P ⊆ NP

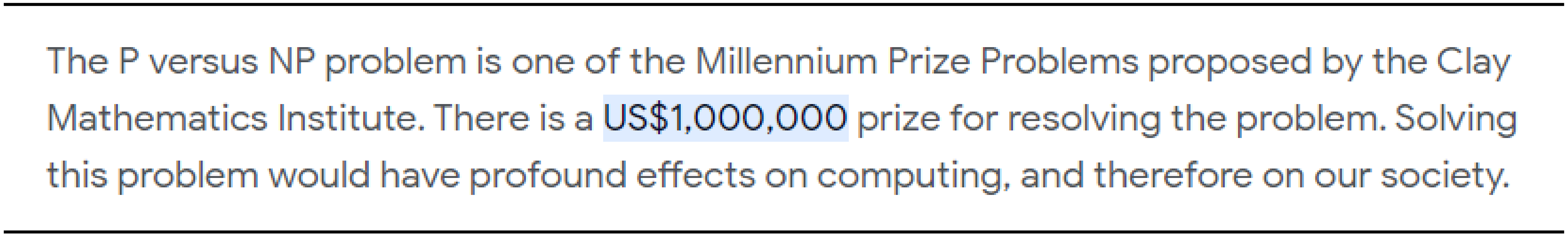


$P \subseteq NP$



P = NP ?

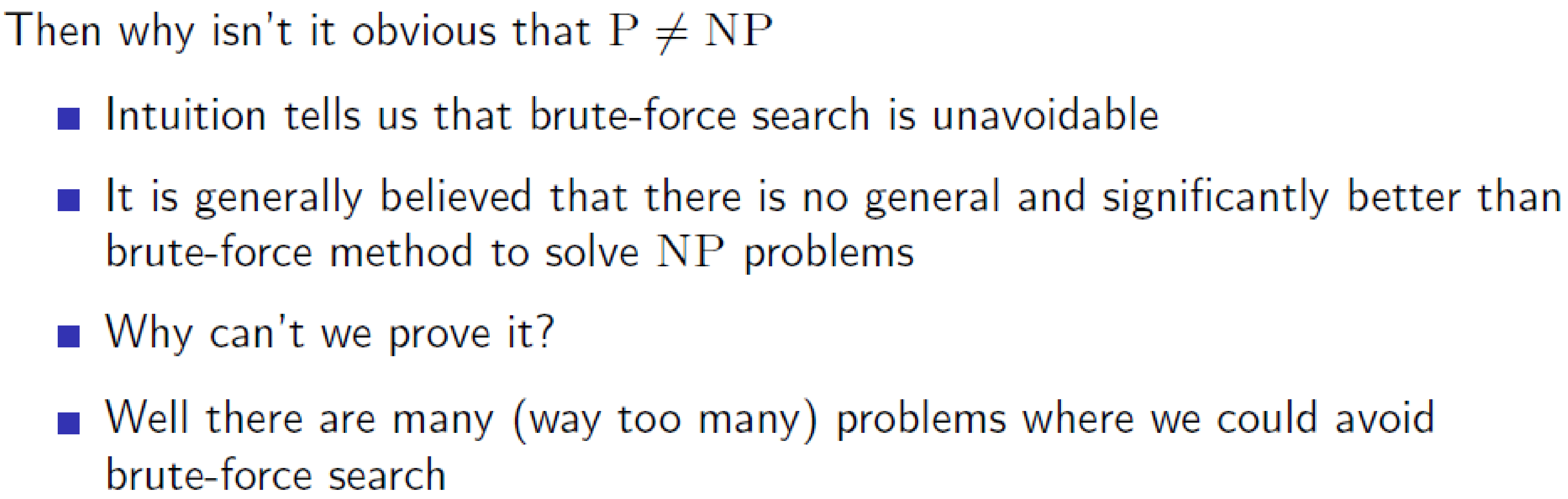
- Many problems in CS, Math, OR, Engineering, etc. are polynomial time verifiable but have no known polynomial time algorithm
- Although polynomial time verifiability seems like a weaker condition than polynomial time solvability, no one has been able to prove that it is weaker
- So it is unknown whether $\mathrm{P} = \mathrm{NP}$
- **The biggest open problem in computer science**



The P versus NP problem is one of the Millennium Prize Problems proposed by the Clay Mathematics Institute. There is a US$1,000,000 prize for resolving the problem. Solving this problem would have profound effects on computing, and therefore on our society.

- No known better way than this
- No proof that there is no better way than this

P = NP ?

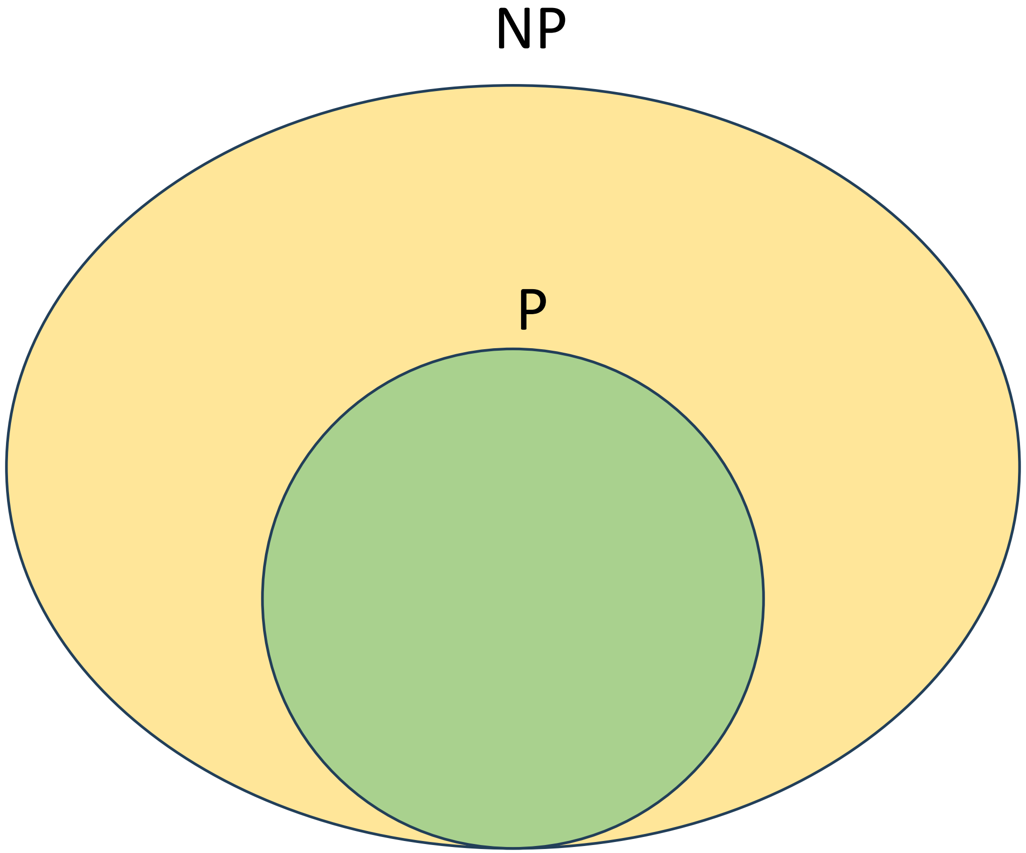


Then why isn't it obvious that $\mathrm{P} \neq \mathrm{NP}$

- Intuition tells us that brute-force search is unavoidable
- It is generally believed that there is no general and significantly better than brute-force method to solve NP problems
- Why can't we prove it?
- Well there are many (way too many) problems where we could avoid brute-force search

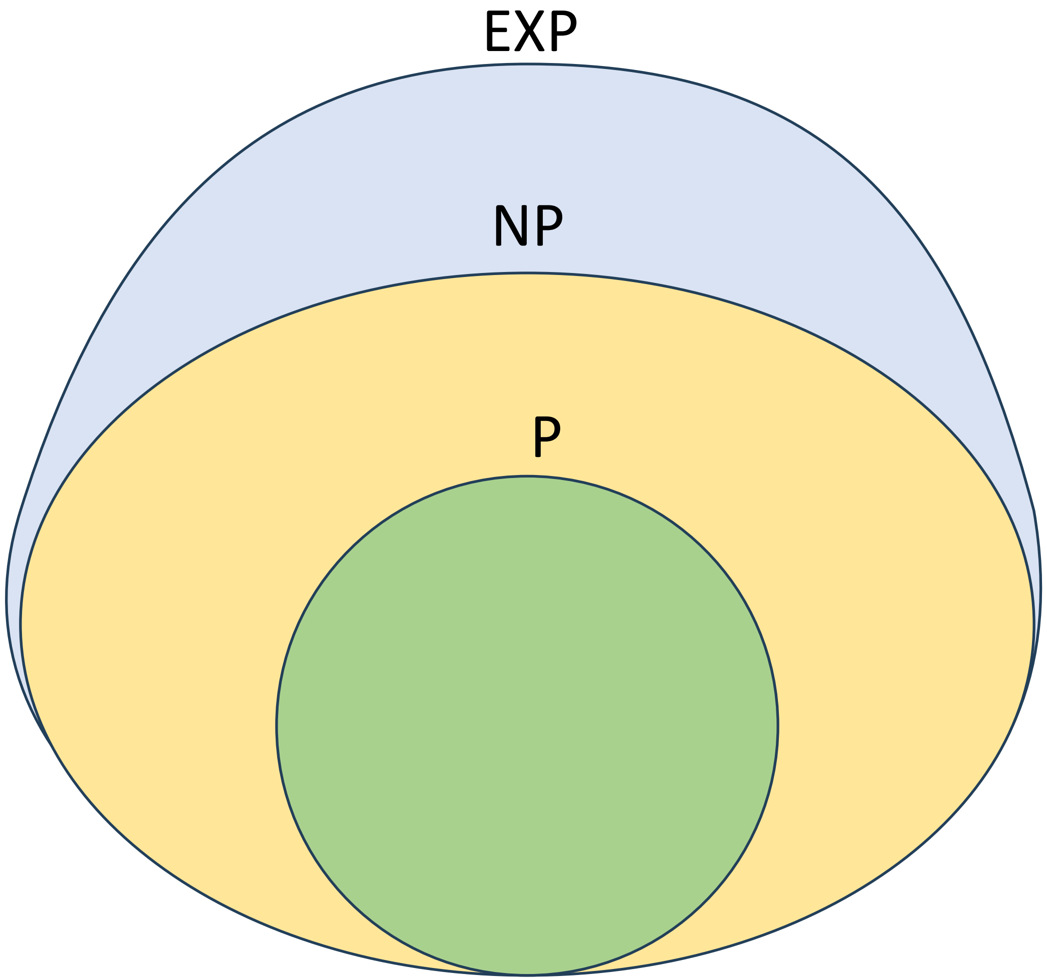
Beyond P and NP

What about problems for which even verifying a solution does not exist in polynomial time?



Beyond P and NP

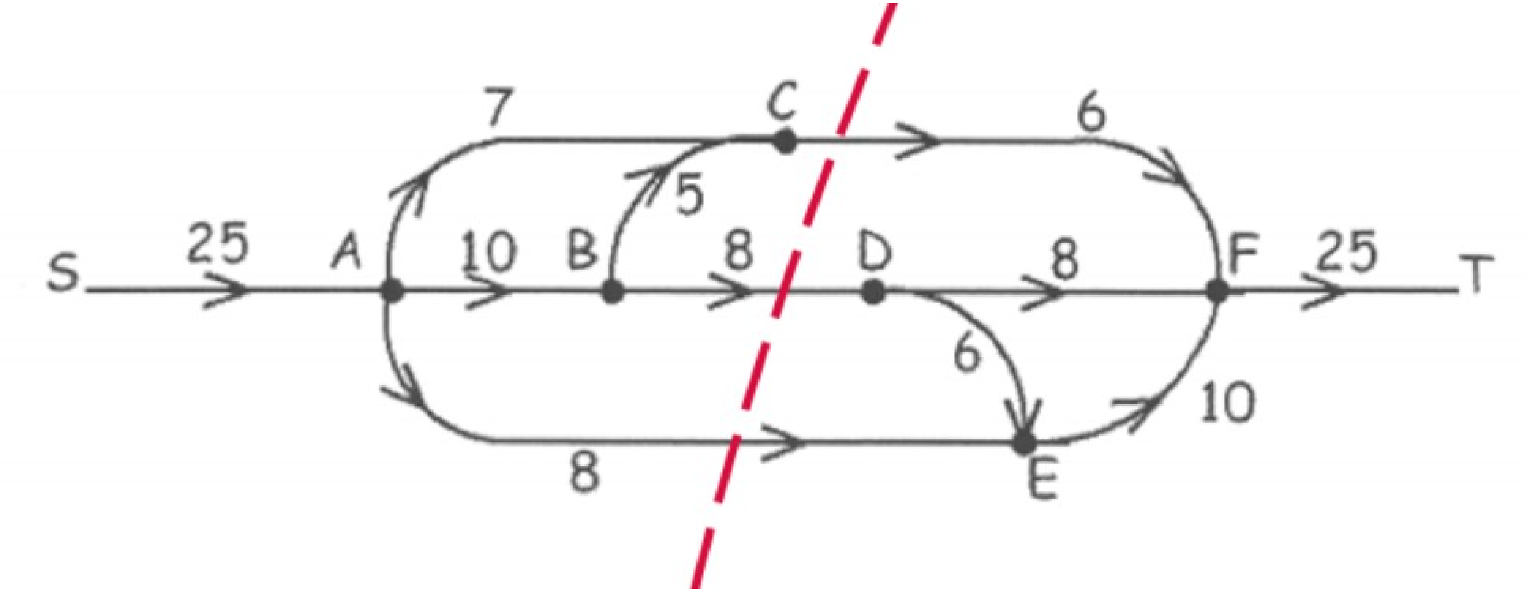
What about problems for which even verifying a solution does not exist in polynomial time?



Quiz 5.5

Surprise Quiz

Quiz 5.5 - Solution



Capacity of the minimum cut=6+8+8=22

Next time

NP Hard

NP Complete

Thanks a lot

If you are taking a Nap, **wake up**........Lecture Over