NP, NP-Complete, NP-hard

Decision vs. Search vs. Optimization

- Optimization problem: find the smallest vertex cover
- Search problem: find a vertex cover of size $\leq k$
- Decision problem: is there a vertex cover of size $\leq k$?

VC-search$\leq_p$ VC-optimization

VC $\leq_p$ VC-Search

VC-Optimization $\leq_p$ VC-Search

Repeatedly call VC-search with different k values (log |V|)

Perform binary search on k

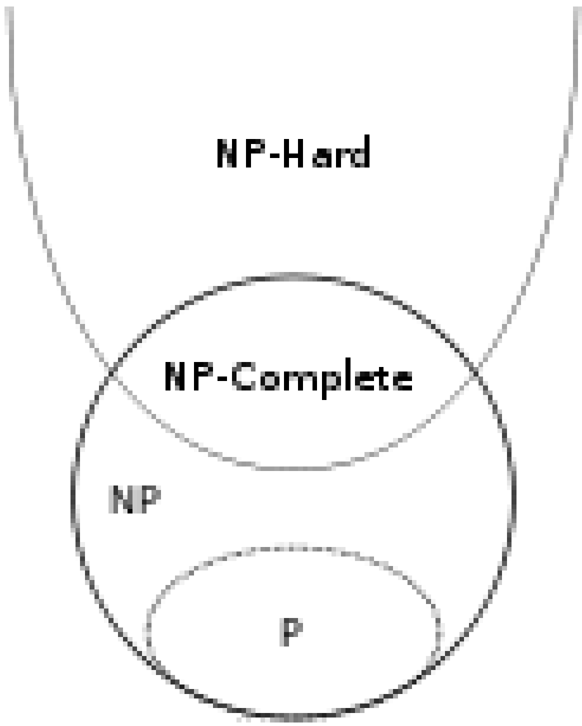
VC-Search $\leq_p$ VC

VC$\equiv_p$VC-Search $\equiv_p$ VC-Optimization

In remainder of this module, we will focus on decision problem

Theory of NP Completeness

Classifying Decision Problems



Assuming that P ≠ NP

P: Class of problems that can be solved in polynomial time

- Corresponds with problems that can be solved efficiently in practice

NP **<u>does not stand for</u>** **"Not Polynomial"**

NP = Nondeterministic Polynomial

What is NP?

- Problems solvable in Non-deterministic Polynomial time . . .

A decision problem is in NP if it satisfies the following:

Given a problem instance I, and any proposed solution S (referred to as a “certificate”) to I, we have an algorithm to verify it that runs in time that is polynomial in |I| (input size of I)

Example NP problems

- Independent set of size K
  - Certificate: an Independent Set S
  - Verification: check each edge $e \in E$ to see if both end points are in S --- $O(|E|)$
- SAT
  - Certificate: A truth assignment to all the variables
  - Verification: check if the formula is satisfied --- O(size of the formula)
- Vertex Cover of size k
  - Certificate: A vertex cover S
  - Verification: check each edge $e \in E$ to see if one of the end points is in S --- $O(|E|)$

NP-Complete (and NP-hard)

- A problem X is NP-complete if
  1. X is in NP
  2. For every Y in NP, $Y \leq_p X$

- X is among the "hardest" problems in NP

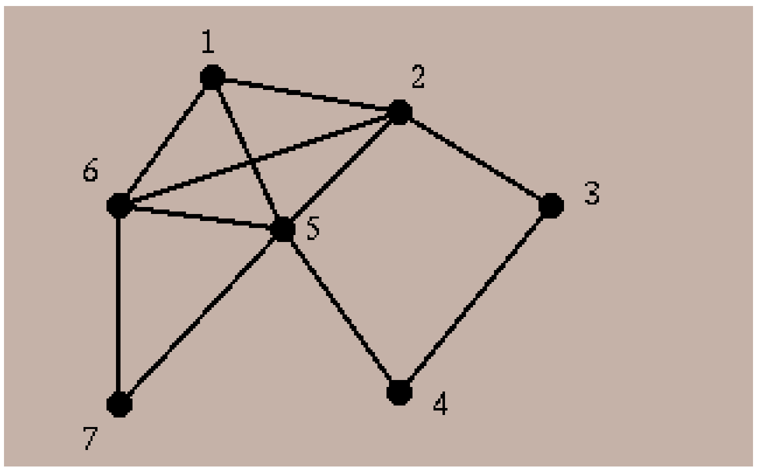
- If a problem satisfies only #2, then it is NP-hard
  - Problems that are at least as hard as all NP problems, but could be harder

Show NP-completeness

- To show a problem Z is NP-complete, we need to show
  1. Z is in NP
  2. Starting from one NP-complete problem X and show that X $\leq_p$ Z
     - Typically start from a problem that is similar to Z
     - Construct a poly-time reduction from X to Z and prove the reduction is correct

Clique

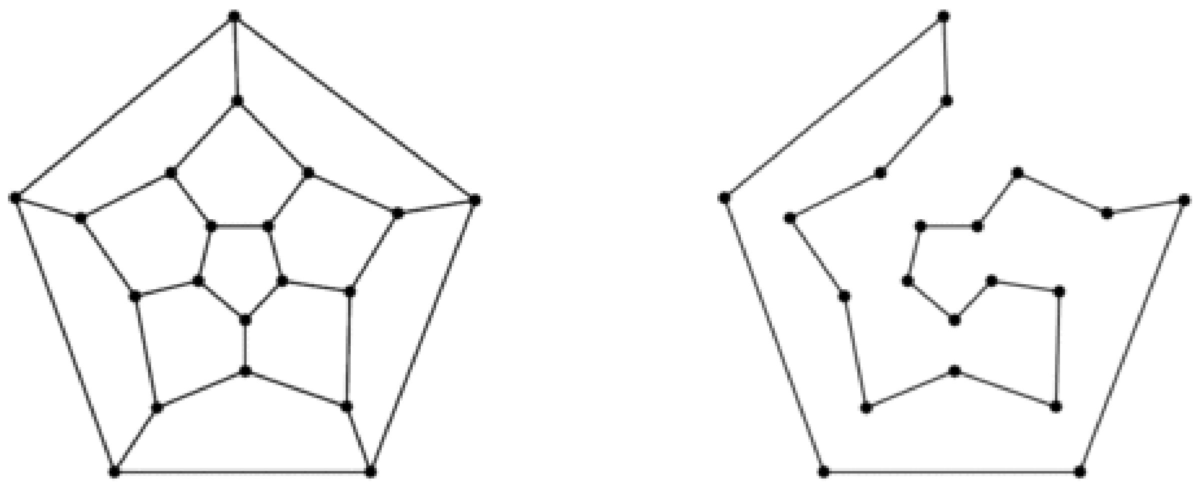
- Given a graph $G = (V,E)$, a clique is a set of nodes $S \subseteq V$ such that every pair of nodes in S there is an edge between them



- Decision problem: given a graph does it contain a clique of size $\geq k$?

Hamiltonian cycle: A known NP-complete problem

- Given a graph, a Hamiltonian cycle is a cycle that visits each node exactly once



- Problem: given a graph does it contain a Hamiltonian cycle?

Traveling salesman problem

- Given a set of cities (nodes in the graph), and a distance d_{ij} between each pair of cities i and j, a total budget D, is there a tour of distance $\leq D$ of all cities that visits each city exactly once and returns to the origin?