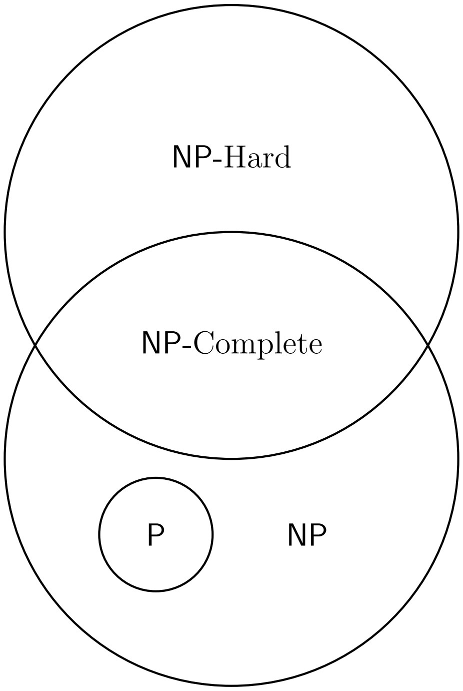
CS 3510: Design & Analysis of Algorithms 11/09/2023

Practice Exam 4: NP-completeness

Abrahim Ladha

- This is the CS 3510 practice exam for exam 4.
- Topics include: complexity classes, polynomial time reductions
- The number of questions on this practice exam is roughly what you could expect on the real exam.



1. For certain, this graph shows a necessarily correct depiction of the relationship between the sets P, NP, NP-Hard, and NP-Complete (this is a trick question).

   ○ True

   ○ False

2. A algorithms researcher has discovered that the running time for an algorithm solving the 3-Coloring problem has a lower bound of $\Omega(2^n)$. If the researcher's discovery is correct, what does that mean in terms of the relationship between P and NP?

   ○ $\mathsf{P} = \mathsf{NP}$

   ○ $\mathsf{P} \neq \mathsf{NP}$

   ○ There is nothing additional that could be said about the relationship between P and NP

3. Let's define the SHORTEST-PATH decision problem to be the following:

   $$\texttt{SHORTEST-PATH} = \{\langle G, s, W, t, k\rangle \mid G \text{ with weights } W \text{ has a path from vertices } s \text{ to } t \text{ of length} \leq k\}$$

   If Alice has found a polynomial time reduction from 3SAT to SHORTEST-PATH, what does that mean in terms of the relationship between P and NP?

   ○ $\mathsf{P} = \mathsf{NP}$

   ○ $\mathsf{P} \neq \mathsf{NP}$

   ○ There is nothing additional that could be said about the relationship between P and NP

4. Every problem in NP$\backslash$P is NP-complete

○ True

○ False

5. You've just made a break through and found a polynomial solution to the LimitSAT! The issue is that it's not exactly the same as normal SAT because each variable can only appear in at most three clauses, and there can only be at most three literals per clauses. Can you show that this is NP-Complete, thus showing $\mathsf{P} = \mathsf{NP}$ (Obviously this is a made up scenario because we don't actually know if $\mathsf{P} = \mathsf{NP}$, but nonetheless, show that LimitSAT is in NP-Complete)?

6. As a student, you're trying to determine your schedule for the upcoming semester. For a particular day, there's some set of classes $s \in S$, each class has some meeting times (a class can meet multiple times in the day i.e. lecture and recitation). You need to determine whether you can schedule k non-overlapping classes throughout the day. Show this problem is NP-Complete.

7. Show that the following `CYCLE-FREE` problem is NP-Complete.

   Input: a directed graph $G = (V, E)$ and an integer k.
   Output: Returns `True` if there exists a set $M \subseteq E$ of size at most k such that $|M| \leq k$ and a subgraph $G' = (V, E \backslash M)$ is cycle-free. Returns `False` otherwise.

8. Show that the `Subgraph-Isomorphism` is in the class NP-complete. The problem is as follows: given as input two undirected graphs G and H, determine whether G is a subgraph of H.

9. Show that the `6-Coloring` problem is in the class NP-complete. The problem is as follows: given as input a graph G, determine whether if there exists an assignment of the vertices to 6 colors such that no edge has both ends the same color.

10. The Frog Software Foundation has recently come out with Knapsack as a service, allowing you to solve the Knapsack decision problem, wherein you input your knapsack problem and a value, and the service will tell you if the best value in Knapsack is above the provided value. How could you use this to efficiently solve the Knapsack optimization problem wherein you're given the Knapsack parameters and you want to find the best possible value. Then discuss your algorithms runtime.