

CSE 200: Spring 2026
Homework 2
due April 27, 2026

Please start your response to each question on a new page.

Question 1 (0-1 Programming is NP-complete). *Linear algebra allows us to efficiently solve problems of the form $Ax \leq b$, where A is a matrix, b is a vector, and the question is whether there exists a vector x such that the equation holds. Formally, if A is an $m \times n$ real-valued matrix with rows $a_1, \dots, a_m \in \mathbb{R}^n$, $b \in \mathbb{R}^m$, we are looking for a solution $x \in \mathbb{R}^n$ such that $\langle a_i, x \rangle \leq b_i$ holds for all $i = 1, \dots, m$.*

We will see that the problem becomes NP-complete if we restrict to binary solutions. Concretely, let A be an $m \times n$ matrix with integer entries and let $b \in \mathbb{Z}^m$. Prove that deciding whether there exists $x \in \{0, 1\}^n$ such that

$$Ax \leq b$$

is NP-complete.

Hint: Reduce from 3SAT.

Academic Integrity Note: *The reductions among Karp's 21 NP-complete problems (e.g., 3SAT, 3COL, CLIQUE, HAMILTONIAN-CYCLE, MAX-CUT, 0-1 IP, etc.) are well-documented and widely available. Do not search for solutions to this specific reduction online before submitting your work.*

Question 2 (Collapses of the Polynomial Hierarchy). *Prove that if for some i it holds that $\Sigma_i^P = \Pi_i^P$, then $\text{PH} = \Sigma_i^P$ (i.e., the polynomial hierarchy collapses to the i^{th} level).*

Question 3 (Co-NP Completeness). *Recall the following definitions:*

- *A language L is in coNP if its complement L^c is in NP.*
- *A language L is coNP-hard if for any language $L' \in \text{coNP}$, there is a polynomial-time reduction from L' to L .*
- *A language L is coNP-complete if it is both in coNP and coNP-hard.*

Prove that a language L is coNP-complete if and only if its complement L^c is NP-complete.

Question 4 (Designing Algorithms in Logspace). Consider the language

$$SUMEQUAL = \{x\#y\#z \mid x, y, z \in \{0, 1\}^*, x + y = z\}.$$

Here, $\#$ is a special separator symbol.

Prove that $SUMEQUAL \in \text{LOGSPACE}$, i.e., $SUMEQUAL$ is computable in logarithmic space.

Question 5. Recall the NL-complete language CONN :

$\text{CONN} = \{(G, s, t) : G \text{ is a directed graph, } s, t \text{ are nodes in } G, \text{ there is a path from } s \text{ to } t\}.$

Assume G has n nodes. There are two families of algorithms to solve CONN :

- BFS/DFS-based algorithms, which use $O(n)$ space and polynomial time.
- Savitch's algorithm (see the notes), which uses $O(\log^2 n)$ space.

Answer the following:

- (a) How much time does Savitch's algorithm require? Does it run in polynomial time? Why or why not? Hint: express the asymptotic time complexity of Savitch.
- (b) If you restrict your algorithm to run in polynomial time, what is the minimal amount of space you can achieve? Can you improve upon the linear space used by BFS/DFS?

For part b, there is no "textbook solution". Instead, I want to see your best effort and creative ideas.