Homework Assignment #1 (100 points, weight 10%)
Due: Thursday, October 18, at 10:00 a.m. (in lecture)

1. (5 points) Asymptotic notation review.
Prove the following properties:
   a. If \( g_1 \in O(f_1) \) and \( g_2 \in O(f_2) \) then \( g_1 + g_2 \in O(f_{\text{max}}) \), where \( f_{\text{max}}(n) = \max\{f_1(n), f_2(n)\} \).
   b. If \( g_1 \in O(f_1) \) and \( g_2 \in O(f_2) \) then \( g_1 \cdot g_2 \in O(f_1 \cdot f_2) \).
   c. \( g \in O(f) \) if and only if \( f \in \Omega(g) \).
   d. \( O(f) = O(g) \) if and only if \( f \in O(g) \) and \( g \in O(f) \).
   e. if \( f \in O(g) \) and \( g \in O(h) \) then \( f \in O(h) \).

2. (15 points) Graph representations review and graph encodings.

Consider the graph:
(Review graph representations in section 22.1 of the textbook (23.1 in 1st edition))
   a. Draw its adjacency matrix representation.
      Using an alphabet containing all decimal digits and additional symbols of your choice (such as ; , ( ) ), give an encoding of the graph above as a string on the alphabet. Show that, for any graph with \( n \) vertices and \( m \) edges, the given encoding will produce a string with length polynomial in \( n + m \).
   b. Do the same 3 steps above, for the adjacency list representation.

3. (20 points) Turing machines, RAM programs and analysis of algorithms.
Consider the language:
\( \text{Even} = \{ x \in \{0,1\}^* : x \text{ is the binary representation of an even number } \} \).
   a. (5 points) Draw the state diagram for a Turing machine \( M_1 \) that decides \( \text{Even} \). Analyse the complexity of \( M_1 \) by giving an asymptotic upper bound (big-Oh notation) for the number of steps taken by the machine on strings of length \( n \).
   b. (8 points) Convert the Turing machine \( M_1 \) found on part a into a RAM program, following the recipe given in the proof of Theorem 2 in pages 47-48 of the lecture notes. Similarly to part a, analyse the complexity of the RAM program.
   c. (7 points) Give a constant time RAM program for deciding \( \text{Even} \). Analyse its complexity, showing that its running time is in \( O(1) \).
4. (15 points) P, NP and co-NP.

We define the complexity class co-NP as: co-NP = \{ L \subseteq \{0,1\}^* : \text{such that } \overline{T} \in \text{NP} \}.

a. (5 points) Prove that the class P, viewed as a set of languages, is closed under union, intersection, concatenation and complement. That is, prove that:

- if \( L_1, L_2 \in \text{P} \), then \( L_1 \cup L_2 \in \text{P} \);
- if \( L_1, L_2 \in \text{P} \), then \( L_1 \cap L_2 \in \text{P} \);
- if \( L_1, L_2 \in \text{P} \), then \( L_1 L_2 \in \text{P} \);
- if \( L \in \text{P} \), then \( \overline{T} \in \text{P} \);

b. (5 points) Prove that \( \text{P} \subseteq \text{co-NP} \).

c. (5 points) Prove that if \( \text{NP} \neq \text{co-NP} \) then \( \text{P} \neq \text{NP} \).

5. (20 points) Decision problems vs search problems.

Show that if \text{HAM-Cycle} \in \text{P} \), then the problem of listing the vertices of a hamiltonian cycle, in order, is polynomial-time solvable. Don’t forget to prove that your algorithm runs in polynomial-time.

(Hint: use the decider for \text{HAM-Cycle} as a subroutine for your algorithm.)

6. (25 points) Proving NP-completeness.

The subgraph-isomorphism problem takes two undirected graphs \( G_1 \) and \( G_2 \) and asks whether \( G_1 \) is isomorphic to a subgraph of \( G_2 \). Show that the subgraph isomorphism problem is NP-complete.

(Hint: Reduce from \text{CLIQUE}.)

Recall the definitions:

We say that a graph \( G' = (V', E') \) is a subgraph of a graph \( G = (V, E) \) if \( V' \subseteq V \) and \( E' \subseteq E \).

Two graphs \( G_1 = (V_1, E_1) \) and \( G_2 = (V_2, E_2) \) are isomorphic if there exists a relabeling of the vertices of \( G_1 \) to vertices of \( G_2 \) maintaining the corresponding edges in \( G_1 \) and \( G_2 \), or more formally, if there exists a bijection \( f : V_1 \rightarrow V_2 \) such that \( \{u, v\} \in E_1 \) if and only if \( \{f(u), f(v)\} \in E_2 \).

IMPORTANT:

Marks will be allotted for conciseness and clarity.

You must read and comply with the policy on plagiarism stated in the course web page.