CompSci 162  Unit 4 Diagnostic Exam 1  Spring 2023

DO NOT OPEN EXAM UNTIL INSTRUCTED TO DO SO

SILENCE AND STORE ALL ELECTRONICS

Format Style: Question Pages / Answer Pages

This is a diagnostic exam intended to help you evaluate your readiness for the real exam.

• On the real test, remember to look for your name ☺

• If you believe a question is ambiguous, write at least two reasonable interpretations and indicate clearly which one you will be using. Then answer your question with that assumption. Unless your interpretation makes the problem much more trivial than intended, we will grade your response as if one of us had made that clarification.

• We will only grade responses marked in the space provided for each question.

• The following problems are \( \mathcal{NP} \)-complete:
  
  – SAT: Given a boolean formula \( \phi \), is it satisfiable?
    
    * Also 3-Sat, where the formula is a conjunction of disjunctions of size three.
  
  – INDEPENDENT SET: Given a graph \( G \) and an integer \( k \), determine if there is some subset of the vertices \( V' \), \( |V'| \geq k \) such that no two vertices in \( V' \) share an edge.

  – VERTEX COVER: Given a graph \( G \) and integer \( k \), determine if there is some subset of the vertices \( V' \), \( |V'| \leq k \) such that each edge is incident to some vertex in \( V' \).

  – DIRECTED HAMILTONIAN PATH: does this graph \( G \) contain a simple path that includes every vertex?
    
    * Also undirected Hamiltonian Path
    * Also Hamiltonian Cycle (directed and undirected)

  – 3-COLOR: Given an undirected graph \( G \), determine whether there is a mapping of vertices in \( G \) to 3 distinct colors such that no edge is monochromatic.

  – SUBSET SUM: Given a set \( S \) of numbers, as well as a number \( T \), determine if there is a subset of \( S \) that sums to exactly \( T \).

• It has been great having you this quarter. ☺
1. (3 points) The minimum spanning tree problem is as follows. We are given a connected, undirected graph $G = (V, E)$ and weights $\{w_e\}$ on the edges. We want to find a spanning tree (a subset of the edges such that if we keep only those edges, it forms a graph that includes every vertex in the original but has no cycles) of minimum total weight.

There are many algorithms for this problem with running times $O(m \log n)$.

(a) Express that optimization problem as a decision problem.

(b) Is the decision version of the problem in $\mathcal{P}$? Why or why not?

(c) Is the optimization version of the problem in $\mathcal{NP}$? Why or why not?

2. (2 points) Recall the Vertex Cover problem: Given an undirected graph $G$ and a positive integer $k$, determine if there is some subset of the vertices $V'$, $|V'| \leq k$ such that for every edge $e = (u, v)$, $u \in V'$ or $v \in V'$ (or both). In lecture, we saw that Vertex Cover is $\mathcal{NP}$-complete and that it has a verifier that runs in time $O(nm)$, where $n$ is the number of vertices in the graph and $m$ is the number of edges.

We can also compute that there are only $\binom{n}{k}$ potentially valid certificates; we can enumerate and attempt to verify these in a total time of $O(m \cdot n^{k+1})$ to decide if a graph has a Vertex Cover of size $k$. Explain why this is or is not a polynomial time algorithm to decide Vertex Cover.
Write your answer to question 1(a) between here and the dotted line.

Write your answer to question 1(b) between here and the dotted line.

Write your answer to question 1(c) between here and the dotted line.

Write your answer to question 2 between here and the end of the page.
Nothing you write on this page will be graded. The next page in this booklet contains a spot to answer these questions. You may use this page as scratch paper if you would like, and room to do so exists.

3. (3 points) We saw in lecture that Independent Set is NP-complete. If the input graph is a tree, however, there is an algorithm to solve this in linear time.

Does the existence of that algorithm prove \( P = NP \)? Why or why not? You \textbf{DO NOT} need to, nor should you, provide the algorithm described.

4. (2 points) A faculty committee has \( n \) members and is going to vote on \( t \) issues during the summer break. On every issue, every professor on the committee can vote “Yes” “no” or abstain (this last one means they do not vote on that issue). Each issue passes if it receives more “yes” votes as it does “no” votes (abstained votes do not count).

The Inner Circle problem is this. The input is a table that tells us how every professor on the committee voted on each issue, along with an integer value \( k \). We say a subset \( P' \) of the professors is an \textit{inner circle} if, for every issue, we can look solely at set \( P' \) to determine the result of the vote – each issue passes if and only if it would have passed had only \( P' \) voted on it.

Show that the problem of determining if there is an inner circle of size \( k \) is NP-complete.
Write your answer to question 3 between here and the dotted line.

Write your answer to question 4 between here and the end of the page.