Final Examination

CS 5114 (Spring 2008)

Assigned: April 30, 2008.
Due: at Torgerson 2160B by 5pm on May 7, 2008.
DO NOT EMAIL YOUR SOLUTIONS TO ME!

Name: ____________________

9-digit PID: ____________________

Instructions

1. For every algorithm you describe, prove its correctness and analyse its running time and space used. I am looking for clear descriptions of algorithms and for the most efficient algorithms you can come up with.

2. If you prove that a problem is $\mathcal{NP}$-Complete, remember to state how long the proof, how long it takes to check the proof, and what the running time of the transformation is.

3. You may consult the textbook, your notes, or the course web site to solve the problems in the examination. You **may not** work on the exam with anyone else, ask anyone questions, or consult other textbooks or sites on the Web for answers. **Do not use** concepts from chapters in the textbook that we have not covered.

4. You must prepare your solutions digitally and submit a hard-copy.

5. I prefer that you use LaTeX to prepare your solutions. However, I will not penalise you if you use a different system. To use LaTeX, you may find it convenient to download the LaTeX source file for this document from the link on the course web site. At the end of each problem are three commented lines that look like this:

```
% \solution{
%
% }
```

   You can uncomment these lines and type in your solution within the curly braces.

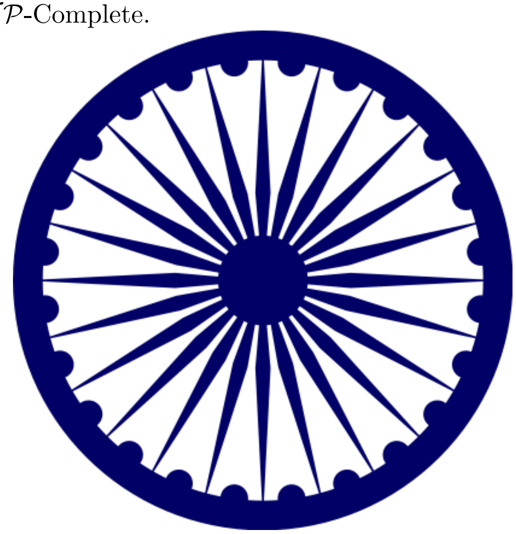
6. Do not forget to staple the hard copy you hand in.

Good luck!

Problem 1 (10 points) Let us start with some quickies. For each statement below, say whether it is true or false.

1. A graph is 2-colourable if and only if it has no cycles of odd length.
2. The harmonic function $H(n) = \Theta(\log n)$.
3. In a directed graph, $G = (V, E)$ let $\pi(u, v)$ denote the length of the shortest path between nodes u and v. Then, for any three nodes u, v, w, $\pi(u, v) + \pi(v, w) \geq \pi(u, w)$.
4. In class, we reduced Independent Set to Vertex Cover. Suppose we have an algorithm that runs in polynomial time and computes a vertex cover that has size at most twice the smallest vertex cover. Then this algorithm yields an independent set of size at least half the largest independent set.
5. Superman gets his powers from the rays of our sun.

Problem 2 (30 points) The flag of a certain populous country contains a symbol called the "Ashoka Chakra" (see the image below). This symbol has a central hub and 24 spokes. Naturally, this reminds us of a graph with 25 nodes and 48 edges, of which 24 nodes are connected by a cycle, and the 25th node is connected to each of the other 24 nodes. A *generalised k-chakra* is a graph with $k + 1$ nodes and $2k$ edges such that k nodes lie on a cycle and the $k + 1$st node is connected to each of the other k nodes. Given an undirected graph G and an integer k, prove that the problem of determining if G contains a generalised k-chakra is $\mathcal{NP}$-Complete.



Problem 3 (30 points) You are given an undirected graph $G = (V, E)$. Each vertex $v \in V$ has a label $l_v \in \{-1, 0, 1\}$. Each edge $e \in E$ has a weight $w_e > 0$. Consider the set V_0 of nodes in V whose label is 0. Your goal is to change the label of every node in V_0 to 1 or -1, while taking the edge structure of G into account. For example, if a node v in V_0 has many more neighbours with label 1 than -1, you would like to change v's label to 1. Therefore, you decide to maximise the *consistency* of G

$$c(G) = \sum_{e=(u,v)\in E} w_e l_u l_v.$$

Either devise a polynomial time algorithm to maximise $c(G)$ or prove that the decision version of the problem (i.e., given a parameter κ, goes G have a labelling of the nodes in V_0 such that $c(G) \geq \kappa$) is $\mathcal{NP}$-Complete.

Problem 4 (30 points) A *Hamiltonian path* in an undirected graph is a simple path that visits every vertex exactly once. Deciding whether a graph has a Hamiltonian path is $\mathcal{NP}$-Complete. Some special graphs (e.g., complete graphs) have simple solutions to the problem. Let G be an undirected graph with nodes $V = \{v_1, v_2, \ldots v_n\}, n \geq 4$. Every node in V has an edge connecting it to $n - 2$ other nodes in V.

You are given the node set V but not the edges of G. The following two functions S and R return information about the edges of G:

S takes two vertices in V as argument and returns `true` if the two vertices are connected by an edge and `false` otherwise.

R takes one vertex v in V as argument and returns the unique node in V (other than v, of course) that is not connected to v.

You may assume that each functions runs in $O(1)$ time. You have two tasks.

1. Prove that G has a Hamiltonian path.
2. Compute the Hamiltonian path in G as a sequence of vertices by using one (but not both) of the following options:

   Option 1 You are allowed to call S but not R. Your algorithm must use $O(1)$ space, $O(n)$ time and at most $n - 1$ calls to S.

   Option 2 You are allowed to call R but not S. Your algorithm must use $O(n)$ space, $O(n)$ time and at most $\lceil n/4 \rceil$ calls to R.