CPSC 420+500 Problem Set 7 26 Mar 2017
W. Evans Due: 3 Apr 2017 at 21.00 (9pm)

Submission Instructions

Handin your solutions using `handin`. You can write your solutions by hand and scan the pages or take pictures of them with your phone; or use a word processing package to typeset your solutions. Whatever you do, you should produce **pdf** files **assign7Q1.pdf**, **assign7Q2.pdf**, etc. containing your solutions for Question 1, Question 2, etc.

To handin: Copy your solution files to the directory `~/cs420/a7` in your home directory on a CS undergraduate machine. (You may have to create this directory using `mkdir ~/cs420/a7`.) Then run `handin cs420 a7` from your home directory.

If you do not have a CS department account, you can email your assign1.pdf to me `will@cs.ubc.ca`.

Late submissions are not accepted.

Grading Policy

We will grade a subset of these questions of size at least two. It's a good idea to do all of the questions because (1) you don't know which ones we'll grade and (2) answering these questions is good practice for the exams. We will email feedback to the email address associated with the account you used to handin the assignment.

Questions

Try to answer these on your own but if you work with someone or use an outside source you must acknowledge them in your write-up. Do not copy solutions from any source.

1. (from Exercise 4 Erickson's Approximation Algorithms notes) Consider the following heuristic for constructing a vertex cover of a connected graph G: Return the set of non-leaf nodes in any depth-first spanning tree of G.

   (a) Prove that this heuristic returns a vertex cover of G.

   (b) Prove that this heuristic returns a 2-approximation to the minimum vertex cover of G.

   (c) Describe an infinite family of graphs for which this heuristic returns a vertex cover of size $2 \cdot \text{OPT}$.

2. (from Problem 9.4 Algorithms by Dasgupta, Papadimitriou, Vazirani) Given an undirected graph $G = (V, E)$ in which each node has degree $\leq d$, show how to efficiently find an independent set whose size is at least $1/(d+1)$ times the size of the largest independent set.

3. (from Exercise 6 Erickson's Approximation Algorithms notes) The chromatic number $\chi(G)$ of a graph G is the minimum number of colors required to color the vertices of the graph, so that every edge has endpoints with different colors. Computing the chromatic number exactly is NP-hard. Prove that the following problem is also NP-hard: Given an n-vertex graph G, return any integer between $\chi(G)$ and $\chi(G) + 573$. [Note: This does not contradict the possibility of a constant factor approximation algorithm.]

4. Show that the FIFO page replacement algorithm is k-competitive.