Approximation Algorithms – COMP 670J
Fall Semester 2001
Homework 1 – Due September 20, 2001

Instructions: Choose ONLY one of the following problems and answer it. Please follow the rules concerning collaboration and plagiarism as laid out in the first class handout. Write your name, email address and ID number clearly on your answer sheet and hand in the homework at the beginning of class on September 21, 1999.

1. Consider the following polynomial time approximation algorithm for Vertex Cover:

   Bad Vertex Cover (BVC)
   $C \leftarrow \emptyset$
   $E' \leftarrow E$
   while E' is not empty
   choose the node v in (V, E') that has largest degree
   (break ties arbitrarily)
   remove from E' every edge touching v.
   return(C)

   Prove that BVC is not a 3-approximation algorithm. That is, give a sequence of graphs $G_i = (V_i, E_i)$ such that $|V_i| \to \infty$ and $\frac{BVC(G_i)}{OPT(G_i)} > 3$.

   (Harder but not necessary: prove that BVC is not a c-approximation algorithm for *any* $c > 0$.

2. (Problem 37.2-1 in CLR) Consider the following **closest-point heuristic** for building an approximate-travelling salesman tour for Metric TSP. Begin with a trivial cycle consisting of a single arbitrarily chosen vertex. At each step, identify the vertex u that is not on the cycle but whose distance to any vertex on the cycle is minimum. Suppose that v is the vertex on the cycle closest to u. Extend the cycle to include u by inserting u just after v. Repeat until *all* vertices are on the cycle. Prove that this heuristic is a 2-approximation algorithm.

3. Bin Packing: (Problem 37-1 in CLR) Suppose that we are given a set of n objects, where the size s_i of the ith object satisfies $0 < s_i < 1$. We wish to pack all the objects into the mimimum number of unit-sized bins. Each bin can hold any subset of the objects whose total size does not exceed 1. This is an NP-Hard problem.

   Now assume that the bins are ordered as first, second, third, etc.. The *first-fit* heuristic takes the objects one at a time and places an object into the *first* bin that currently has space for it. Let $S = \sum_{i=1}^{n} s_i$.

   - Prove that the optimal number of bins required is at least $\lceil S \rceil$.
   - Prove that the number of bins used by the first-fit heuristic is never more than $\lceil 2S \rceil$.
   - Prove that *first-fit* is a 2-approximation algorithm.