Algorithms Qualifying Exam, January 2012

Computer Science Department

Michigan Technological University

Instructions:

1. This is a closed-book exam. You are not allowed to use the textbook, your class notes or lectures.
2. Please solve 5 problems out of 6. All problems are equally weighted. Please read all problems carefully and start with the problems you think are easier to solve.
3. You have 3 hours for this exam.
4. Please be precise and concise in your writing. It is important that you clearly present your solutions.
5. Unless a problem explicitly asks, you may reuse well-known algorithms or theorems without their proof of correctness.
6. For all problems that ask for the design of an algorithm, you should prove the correctness of your algorithms. Please, make your proof explicit in your solution.

1. Please complete the following two questions:

   (a) Let A be a list of n (not necessarily distinct) integers. Describe an $O(n)$-algorithm to test whether any item occurs more than $\lceil n/2 \rceil$ times in A.

   (b) Describe an $O(n)$-algorithm that, given a set A of n distinct numbers and a positive integer $k \leq n$, determines the k numbers in A that are closest to the median of A.

2. Consider the following sorting algorithm:

```
SLOW-SORT(A, i, j)
  if A[i] > A[j]
     then exchange A[i] and A[j]
  if i+1 >= j
     then return
   k = floor((j-i+1)/3)
   SLOW-SORT(A, i, j-k)  // first two-thirds
   SLOW-SORT(A, i+k, j)  // last two thirds
   SLOW-SORT(A, i, j-k)  // first two-thirds again
```

   Argue that SLOW-SORT(A, 1, n) correctly sorts the input array A[1..n], and give a tight asymptotic bound for its running time.

3. You have a set of n integers each in the range $[0..K]$. Partition these integers into two subsets such that you minimize $|S_1 - S_2|$, where S_1 and S_2 denote the sums of the elements in each of the two subsets.

   (a) Design a *dynamic programming* algorithm to solve this problem.

   (b) Analyze the complexity of your proposed algorithm.

4. The edge connectivity of an undirected graph is the minimum number of edges that must be removed to disconnect the graph. Show how the edge connectivity of an undirected graph $G =< V, E >$ can be determined by running a maximum-flow algorithm on at most $|V|$ vertices and $O(E)$ edges.

5. Suppose that we are given a set of n objects, where the size s_i of the i_{th} object satisfies $0 < s_i < 1$. We wish to pack all the objects into the minimum number of unit-size bins. Each bin can hold any subset of the objects whose total size does not exceed 1.

   (a) Prove that the (bin-packing) problem of determining the minimum number of bins required is NP-hard by reducing from set-partition problem which is NP-complete. The set-partition problem determines if a set of numbers S can be partitioned into two sets A and $S - A$ such that the sum of elements in A is equal to the sum of the elements of $S - A$.

   (b) Find a polynomial-time two-approximation algorithm for the bin-packing problem.

6. In the Three-Coloring Problem, we need to determine whether the vertices of a giving undirected graph can be painted with three colors, in such a way that no two adjacent vertices have the same color. Similarly, in the Four-Coloring Problem, we need to determine whether the vertices can be painted with four colors.

   (a) The Three-Coloring Problem is known to be NP-complete. Use this fact to prove that the Four-Coloring Problem is also NP-complete.

   (b) Extend your proof in (a) to show that the 2012-Coloring Problem is also NP-complete.