Department of Applied Mathematics, National Chiao Tung University
PhD Qualifying Exam for Fall 2007 --- Algorithms

96年PA

★There are 7 questions in this exam. For every question, please write your answer in a clean and concise way. If you are asked to write an algorithm for a question, you have to neatly write the pseudo-code of your algorithm and also put explanation about your pseudo-code.

1. (20%) 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 minimum number of unit-size bins. Each bin can hold any subset of the objects whose total size does not exceed 1. The ***first-fit heuristic*** takes each object in turn and places it into the first bin that can accommodate it. Let $S = \sum_{i=1}^{n} s_i$.

(a) Argue that the optimal number of bins required is at least $\lceil S \rceil$.

(b) Argue that the first-fit heuristic leaves at most one bin less than half full.

(c) Prove that the number of bins used by the first-fit heuristic is never more than $\lceil 2S \rceil$.

(d) Prove an approximation ratio of 2 for the first-fit heuristic.

2. (10%) Suppose we tried merge sort with four pieces rather than two. Then we have

$$T(n) = \begin{cases} 4T(\lfloor \frac{n}{4} \rfloor) + cn & \text{if } n > 4, \\ c & \text{if } n \le 4. \end{cases}$$

Show that there exist constants a and n_0 such that $T(n) \leqq an \log(n)$ for all $n \geqq n_0$.

3. (15%) Analyze the running time of the following recursive procedure as a function of n. You may assume that each assignment or division takes unit time. Please give your answers by using asymptotic "Θ" analysis.

```
Procedure NCTU(int n)
   if n<2 then return;
   else
      count:= 0;
      for i:= 1 to 8 do
         NCTU(n div 2);
      for i:= 1 to n³ do
         count:= count + 1;
```

4. (15%) Consider the ***mergesort*** algorithm for sorting a set of n points.

(a) Draw the recursion tree for this algorithm for n=13.

(b) How many levels are there in the recursion tree?

(c) How many comparisons are done at each of the levels in the worst case?

(d) What is the total number of comparisons needed?

(e) Generalize your results for parts (b-d) for arbitrary n (you man assume that n is a power of 2). Please give your answers using the big-oh notation.

5. (10%) A ***unit-length closed interval*** on the real line is an interval $[x, 1+x]$. Describe an $O(n)$ algorithm that, given input set $X=\{x_1, x_2, \ldots, x_n\}$, determines the smallest set of unit-length closed intervals that contains all of the given points. Argue that your algorithm is correct. You should assume that $x_1 < x_2 < \ldots < x_n$.

6. (10%) (a) Prove that $P \subseteq$ co-NP. (b) Prove that if NP$\neq$co-NP then P$\neq$NP.

7. (20%) Given an integer m-by-n matrix A and an integer m-vector b, the ***0-1 integer programming problem*** asks whether there is an integer n-vector x with elements in the set $\{0,1\}$ such that $Ax \leqq b$. Prove that 0-1 integer programming is **NP**-complete. (Hint: Reduce from 3-CNF-SAT.)