CSC 238F 1998

Homework Assignment 2, Due Oct. 21

On the cover page of your assignment, you must write **and sign** *the following statement: "I have read and understood the policy on collaboration in homework stated in the Course Information handout." Without such a signed statement your homework will not be marked. On the cover page, please inlcude your name, student number and tutorial section (e.g. the room and/or the name of your TA.)*

---

1. Write a piece of pseudo-code that satisfies the following precondition/postcondition pair, and *prove* carefully that it is correct. Your program should be "essentially" a binary search, so that it runs in time $O(\log n)$.
   Note: In order to get more than 0 marks for this question, it is necessary, but not sufficient, that your program be completely correct.

   Precondition:
   n is an integer, $n \geq 1$.
   A is an integer array, sorted in reverse order between positions 1 and n. That is, For all i, j such that $1 \leq i \leq j \leq n$, $A(i) \geq A(j)$. (Note that A may contain repeated values.)
   x is an integer.

   Postcondition: n, x and A have not been changed.
   If there exists an i such that $1 \leq i \leq n$ and $A(i) > x$, then the program prints the *largest* such value of i; if there does not exist such an i, then the program prints "failure".

2. The following is a piece of pseudo-code for approximating the square root of a natural number x. The code assumes we start with a natural number s that is greater than $\sqrt{x}$, and it computes the number $r = \lfloor \sqrt{x} \rfloor$.
   Prove carefully that the following pseudo-code is correct with respect to the following precondition/postcondition pair. (Note that by $\lfloor y \rfloor$ we mean the largest integer $\leq y$, and by $\lceil y \rceil$ we mean the smallest integer $\geq y$.)

   ```
   r = 0;
   LOOP
       while (s ≠ 1)
       s = ⌈s/2⌉;
       if (r + s) * (r + s) ≤ x
           r = r + s
   END LOOP
   ```

   Precondition:
   $x \in \mathbb{N}$ and $s \in \mathbb{N}$ and $s^2 > x$.

   Postcondition: $r = \lfloor \sqrt{x} \rfloor$.
   That is, $r \geq 0$ and $r^2 \leq x$ and $(r+1)^2 > x$.

3. We wish to determine the time complexity of the program in question 2. Let us assume that the code is only executed when s is the smallest (integral) power of 2 greater than $\sqrt{x}$. Let $T(n)$ be the largest possible time the program can take, when x is a number with at most n bits (that is, $x < 2^n$). In order to get an upper bound on $T(n)$, we have to know how long it takes to perform the tests and arithmetic statements in the program. Let us assume that comparisons, additions and subtractions on numbers that each have at most m bits, are performed within time $O(m)$; let us assume that multiplications and divisions on numbers that each have at most m bits, are performed within time $O(m^2)$.

   Show that $T(n) \in O(n^3)$.

4. (a) Prove that if $f \in O(g)$ and $g \in O(h)$, then $f \in O(h)$.

   (b) prove that for arbitrary functions f and g, $g \in O(f)$ if and only if $f \in \Omega(g)$.
   NOTE: It is important to realize that there are *two* things to prove here. You have to show:
   $g \in O(f) \Rightarrow f \in \Omega(g)$ and
   $f \in \Omega(g) \Rightarrow g \in O(f)$.

   (c) Let $f(n) = n^5/30$ and $g(n) = 45n^4$. Prove that $f \notin O(g)$.