CS 573: Algorithms, Fall 2012
Homework 0, due Monday, September 3, 23:59:59, 2012

Name:	
Net ID:	

Neatly print your name (first name first, with no comma), your network ID, and a short alias into the boxes above. **Do not sign your name. Do not write your Social Security number.** Staple this sheet of paper to the top of your homework.

This homework tests your familiarity with the prerequisite material from CS 173, CS 225, and CS 373—many of these problems have appeared on homeworks or exams in those classes—primarily to help you identify gaps in your knowledge. **You are responsible for filling those gaps on your own.** Chapters 1–6 of CLR should be sufficient review, but you may want to consult other texts as well.

Before you do anything else, read the Homework Instructions and FAQ on the CS 573 course web page (`http://courses.engr.illinois.edu/cs573/fa2012/faq/`), and then check the box below. This web page gives instructions on how to write and submit homeworks—staple your solutions together in order, write your name and netID on every page, don't turn in source code, analyze everything, use good English and good logic, and so forth.

☐ I have read the CS 573 Homework Instructions and FAQ.

Remember to do the quiz on moodle!

"Be that as it may, it is to night school that I owe what education I possess; I am the first to own that it doesn't amount to much, though there is something rather grandiose about the gaps in it." – The tin drum, Gunter Grass

Required Problems

1. MOVING NUMBERS. [**50 Points**]

   (A) [**25 Points**] The input is a multiset X of n positive integer numbers in the range 1 to k. Consider the famous algorithm:

   ```
   Play(X) :
       while X contains more than one element do
           if X contains the number 0 then
               Remove the number 0 from X
               continue
           Two distinct elements x₁ and x₂ are chosen arbitrarily from X
           y₁ = min(x₁, x₂) − 1
           y₂ = max(x₁, x₂) + 1
           X ← (X \ {x₁, x₂}) ∪ {y₁, y₂}
   ```

   Here is an example of the execution of **Play**($\{1, 2, 3, 4\}$).

   $$\{1, 2, 3, 4\} \implies \{1, 2, 2, 5\} \implies \{0, 2, 3, 5\} \implies \{2, 3, 5\} \implies \{2, 2, 6\}$$
   $$\implies \{3, 1, 6\} \implies \{1, 2, 7\} \implies \{1, 8, 1\} \implies \{0, 2, 8\}$$
   $$\implies \{2, 8\} \implies \{1, 9\} \implies \{0, 10\} \implies \{10\}.$$

   Prove (maybe using induction, but you do not have to) that **Play** always terminates. (Hint: Come up with an argument why in each step some non-trivial progress is being made.)

   (B) [**25 Points**] (Harder.) Prove that the algorithm always terminates after $O\big((kn)^2\big)$ steps.

   Extra fun questions: Since this problem is so much "fun", here are a few more questions. Do not submit a solution for this part (we also will not provide solutions to this part).

   (I) (Hard.) Prove that, in the worst case, $\Theta\big((kn)^{3/2}\big)$ steps are needed (that is, improve the upper bound, and provide a matching lower bound) if $k \leq n$. For $k > n$, prove that the right bound is $\Theta\big(kn^2\big)$.

   (II) Show that if instead of increasing the maximum number by one, we increase it by two, then the number of steps in the worst case can be exponential.

   (III) (Easy.) Consider the modified algorithm, where the algorithm repeatedly takes a subset of numbers (out of the current set of numbers), decrease the smallest number in the subset by 1, and change the value of all the other numbers in an arbitrary fashion (again, we remove a number when it becomes zero). Prove that this algorithm always terminates.

2. ROLLING TO VICTORY. [**30 Points**]

Alice and Bob each have a fair n-sided die. Alice rolls her die once. Bob then repeatedly throws his die until he rolls a number at least as big as the number Alice rolled. Each time Bob rolls, he pays Alice $1. (For example, if Alice rolls a 5, and Bob rolls a 4, then a 3, then a 1, then a 5, the game ends and Alice gets $4. If Alice rolls a 1, then no matter what Bob rolls, the game will end immediately, and Alice will get $1.)

Exactly how much money does Alice expect to win at this game? Prove that your answer is correct. If you have to appeal to 'intuition' or 'common sense', your answer is probably wrong!

3. RANDOM WALK. [**20 Points**]

A ***random walk*** is a walk on a graph G, generated by starting from a vertex $v_0 = v \in V(G)$, and in the ith stage, for $i > 0$, randomly selecting one of the neighbors of v_{i-1} and setting v_i to be this vertex. A walk $v_0, v_1, \ldots, v_m$ is of length m.

(A) For a vertex $u \in V(G)$, let $P_u(m, v)$ be the probability that a random walk of length m, starting from u, visits v (i.e., $v_i = v$ for some i).
Prove that a graph G with n vertices is connected, if and only if, for any two vertices $u, v \in V(G)$, we have $P_u(n - 1, v) > 0$.

(B) Prove that a graph G with n vertices is connected if and only if for any pair of vertices $u, v \in V(G)$, we have $\lim_{m \to \infty} P_u(m, v) = 1$.