CS 577: Introduction to Algorithms	Homework 0
Out: 09/5/06	Solutions: 09/12/06

Ground Rules

This is a self-calibration assignment. You should use this to refresh your knowledge of the prerequisites for this course, and to get an idea of the level of difficulty of this course. This assignment will not be graded but model solutions will be handed out in class on 9/12.

Reading

Review Chapter 2 of the course text. This is an important part of the homework, and although we will not review this material in class, future topics rely heavily on a good understanding of it.

Problems

1. **(Treasure Hunt.)** You are in the middle of a long highway when you find a map to an ancient treasure. The treasure is apparently hidden under one of the milestones along the highway. Unfortunately, there is no way of knowing how far from your current position the treasure is hidden, or whether it is to your left or to your right. The only way to find whether or not the treasure is at a certain milestone is to find a small marker in the sand beside the correct milestone. You can search for the treasure by going left or right and changing your direction as many times as you want.

   (a) Devise a search strategy that is guaranteed to find the treasure. Assume that you walk at a speed of 1 mile an hour. How much time do you take to find the treasure if it is at a distance of n miles from your current position? (You answer should be in terms of n.)

   (b) A local bully happens to find the treasure map at the same time and location as you. He sends two of his minions, one in each direction, to locate the treasure. The minions walk at a speed of 1 mile an hour. Luckily, you have a bicycle at your disposal. How fast do you have to ride your bike so as to find the treasure before the bully finds it? (As before, the treasure is at an unknown distance of n miles from your starting point. Your answer should be independent of n or the length of the highway.)

2. **(The Towers of Hanoi)** An ancient temple at Hanoi, Vietnam contains a large room with three towers in it surrounded by n golden discs[1] of different sizes. The priests at the temple, acting out the command of an ancient prophecy, have been moving these discs in accordance with the following rules: (1) the discs must be moved one at a time from one pole to another, and, (2) a disc can never be placed over a smaller disc. At the beginning of time, the first tower contained all the discs, the largest at the bottom and the smallest at the top. Legend has it that when all the discs are moved to the third tower, the world will end.

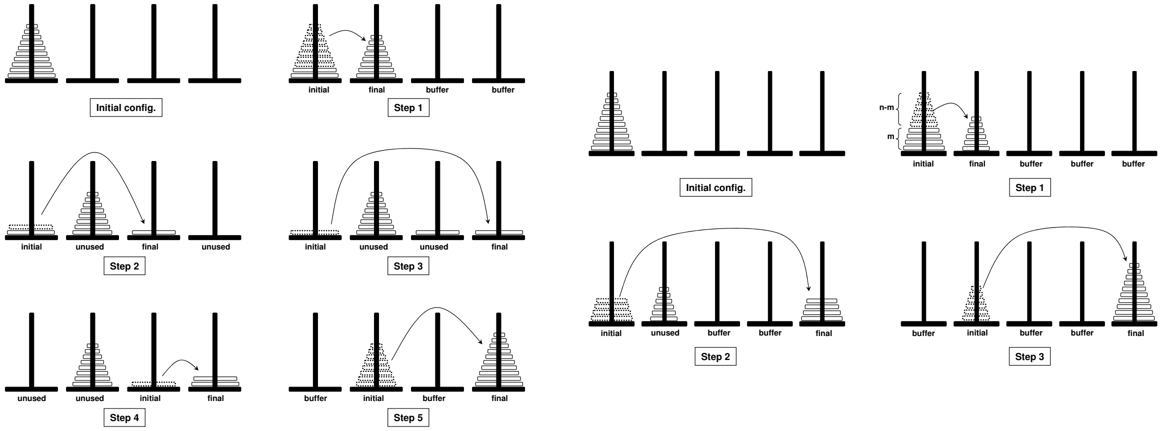
   (a) What is the smallest number of steps in which the priests can transfer all the discs from the first tower to the third? (**Hint:** Consider the process that recursively transfers the top $n-1$ discs to the second tower, then the largest disc to the third tower and then the top $n-1$ discs from the second tower to the third.)

   (b) We will now figure out what happens if the priests have one extra tower to use. The goal, as before, is to transfer the discs from the first tower to the last, using the two intermediate ones as buffers. Consider the following simple strategy. The priests recursively transfer $n-2$ discs to the second tower, using the third and fourth as buffers. They then transfer the remaining two discs to tower 4, using tower 3, as shown in Figure 1(a) (Steps 2–4). Finally, they recursively transfer the $n-2$ smaller discs from tower 2 to tower 4 using towers 1 and 3 as buffers.

   (i) Suppose that $T(n)$ denotes the time taken to transfer the n discs from one tower to another using two intermediate towers. Write an expression for $T(n)$ in terms of $T(n-2)$.

   (ii) Solve for $T(n)$. (Note that your answer would be slightly different for odd and even n.)

[1] In the original problem, the number of discs is 64.



(a) (b)

Figure 1: The Towers of Hanoi: (a) four towers; (b) five towers

(iii) What factor of improvement in the number of steps do the priests obtain by using four towers instead of three when $n = 8$? What if $n = 64$?

(c) We now extend the above argument to five towers. Consider the following strategy for the priests. The priests select some number $m < n$. They then recursively transfer the smallest $n - m$ discs to the second tower, using the third, fourth and fifth as buffers. Then they transfer the remaining m discs to tower 5 using only the towers 3 and 4 as buffers (see Figure 1(b)). Finally, they transfer the smallest $n - m$ discs from tower 2 to tower 5 using towers 1, 3 and 4 as buffers.

(i) Recall that $T(n)$ is the number of steps required to move n discs from one tower to another using **two** buffer towers, as computed in part (b). Let $S(n)$ denote the number of steps required to move n discs from one tower to another with **three** buffer towers using the strategy described above, where m is some fixed number. Write a recursive expression for $S(n)$ in terms of $T(m)$ and $S(n - m)$.

(ii) Solve this recurrence and find an expression for $S(n)$ in terms of m and n. Use your answer to part (b) for this problem. (**Hint:** Suppose that $T(m)$ is some constant x. Solve the recurrence in terms of x, n and m. Alternately, solve the problem first for $m = 2, 3, 4$ etc. Guess a solution for general m and prove that it is correct.)

(iii) What is the value of m in terms of n that minimizes the number of steps taken? What is the corresponding value of $S(n)$? (An exact answer is not required; you can use Big-Oh notation for both the expressions.)

3. **(The Winners' Circle.)** n players compete in a chess tournament; every player plays every other player exactly once, and there are no draws. A *winners' circle* is a set of players W such that every other player is beaten by at least one player in W. Prove that regardless of the outcomes of the games, there exists a winners' circle of size at most $\log n$. Give a polynomial time algorithm to find such a set.

4. **(Asymptotics)** Suppose we have three functions f(n), g(n), and h(n) such that f(n) = O(h(n)) and g(n) = O(h(n)). Must it be the case that f(n) = O(g(n))? Explain why or give a counterexample showing why not.