CS 161 Summer 1997

Handout #23

Problem Set 6 Due 3:05p, Tuesday, August 5, 1997

Readings

1. Handout 25.
2. Section 37.3 (skip the analysis section).
3. Chapter 18.
4. *Reminder:* this is the last problem set; after you turn it in, please concentrate on your project. By 5:00p, Friday, August 15, 1997, you should turn in *all* your assignments, including your problem sets and your project. There will be absolutely *no extensions* to this deadline.

Exercises

Exercises are for you to work out, with the assistance of classmates and the course staff. Please don't hand in your solutions.

1. Exercise 18.1-1.
2. Exercise 18.2-1.
3. Exercise 18.3-3.
4. Exercise 18.4-3.
5. Show that the amortized time of a zig-zig splay is at most

   $$3(R_f(X) - R_i(X)).$$

   This fills in the missing details of the proof in handout 25.

Problems

Your problem set grade depends only on the solutions you hand in for the problems below.

1. **[20 points]** Exercise 18.2-3.
2. **[10 points]** Exercise 18.4-4.
3. **[40 points]** Suppose there exist two stacks called A and B, onto which we may apply any of the following operations; below, $size[S]$ is the number of elements in stack S immediately prior to the operation:

- PUSH-A(d): pushes a datum d onto stack A; the real cost of this operation is 1.
- PUSH-B(d): pushes a datum d onto stack B; the real cost of this operation is 1.
- MULTIPOP-A(k): removes $m = \min(k, size[A])$ elements from stack A; the real cost of this operation is m.
- MULTIPOP-B(k): removes $m = \min(k, size[B])$ elements from stack B; the real cost of this operation is m.
- TRANSFER(k): repeatedly pops elements from stack A and pushes them onto stack B, until either k elements have been moved, or A is empty; the real cost of this operation is equal to the number of elements moved. *Tip:* you can only transfer elements *from A to B.*

Please perform the following tasks:

(a) **[15 points]** Assign amortized costs to each operation using the accounting method. Using these costs, show an $O(n)$ worst-case bound on the cost of n operations, starting with A and B both empty.

(b) **[15 points]** Design a potential function such that the amortized cost of each operation is (bounded from above by) a constant, and evaluate this constant for each operation.

(c) **[10 points]** Using your potential function, show an $O(n)$ worst-case bound on the cost of n operations, starting with A and B both empty.

4. **[30 points]** Problem 18-3.