Assignment 6

Due: March 3, 2020

Your solutions must be typed (preferably typeset in LaTeX) and submitted as a hard-copy at the beginning of class on the day its due.

Problem 1: Longest-Probe Bound for Hashing Suppose we use an open-addressed hash table (section 11.4 in CLRS) of size m to store $n \leq \frac{m}{2}$ items.

(a) [10 points] Assuming simple uniform hashing, show that for $i = 1, 2, ..., n$, the probability is at most 2^{-k} that the ith insertion requires strictly more than k probes.

(b) [10 points] Assuming simple uniform hashing, show that for $i = 1, 2, ..., n$, the probability is $O(\frac{1}{n^2})$ that the ith insertion requires more than $2 \lg n$ probes.

Problem 2: Building a Queue using Stacks It is possible to build a *queue* (FIFO) using two stacks. Assume that the stacks have three operations, *push*, *pop*, and *isEmpty*, each with cost 1. A queue can be implemented as follows:

- *enqueue:* push item x onto stack 1
- *dequeue:* if stack 2 is empty then pop the entire contents of stack 1 pushing each element in turn onto stack 2. Now pop from stack 2 and return the result.

A conventional worst-case analysis would establish that *dequeue* takes $O(n)$ time, but this is clearly a weak bound for a sequence of operations, because very few dequeues will actually take that long. To simplify your analysis only consider the cost of the push and pop operations.

(a) [10 points] Using the aggregate method show that the amortized cost of each *enqueue* and *dequeue* is constant.

(b) [10 points] Using the accounting method show that the amortized cost of each *enqueue* and *dequeue* is constant.

(c) [10 points] Using the potential method show that the amortized cost of each *enqueue* and *dequeue* is constant.