New CS 473: Algorithms, Spring 2015
Homework 5

Due Tuesday, March 10, 2015 at 5pm

All homework must be submitted electronically via Moodle as separate PDF files, one for each numbered problem. Please see the course web site for more information.

---

1. On their long journey from Denmark to England, Rosencrantz and Guildenstern amuse themselves by playing the following game with a fair coin. First Rosencrantz flips the coin over and over until it comes up tails. Then Guildenstern flips the coin over and over until he gets as many heads in a row as Rosencrantz got on his turn. Here are three typical games:

   Rosencrantz: **H H** T
   Guildenstern: H T **H H**

   Rosencrantz: T
   Guildenstern: (no flips)

   Rosencrantz: **H H H** T
   Guildenstern: T H H T H H T H T T **H H H**

   (a) What is the *exact* expected number of flips in one of Rosencrantz's turns?

   (b) Suppose Rosencrantz happens to flip k heads in a row on his turn. What is the *exact* expected number of flips in Guildenstern's next turn?

   (c) What is the *exact* expected total number of flips (by both Rosencrantz and Guildenstern) in a single game?

   Include formal proofs that your answers are correct. If you have to appeal to "intuition" or "common sense", your answer is almost certainly wrong!

2. Recall from class that a ***priority search tree*** is a binary tree in which every node has both a *search key* and a *priority*, arranged so that the tree is simultaneously a binary search tree for the keys and a min-heap for the priorities. A ***heater*** is a priority search tree in which the *priorities* are given by the user, and the *search keys* are distributed uniformly and independently at random in the real interval $[0, 1]$. Intuitively, a heater is an "anti-treap".

   The following questions consider an n-node heater T whose priorities are the integers from 1 to n. Here we identify each node in T by its ***priority rank***, rather than by the rank of its search keys; for example, "node 5" means the node in T with the 5th smallest *priority*. In particular, the min-heap property implies that node 1 is the root of T. Finally, let i and j be arbitrary integers such that $1 \le i < j \le n$.

(a) What is the *exact* expected depth of node j in an n-node heater? Answering the following subproblems will help you:

i. Prove that in a uniformly random permutation of the $(i+1)$-element set $\{1, 2, \ldots, i, j\}$, elements i and j are adjacent with probability $2/(i+1)$.

ii. Prove that node i is an ancestor of node j with probability $2/(i+1)$. *[Hint: Use the previous question!]*

iii. What is the probability that node i is a *descendant* of node j? *[Hint: Do **not** use the previous question!]*

(b) Describe and analyze an algorithm to insert a new item into a heater. Express the expected running time of the algorithm in terms of the priority rank of the newly inserted item.

(c) Describe an algorithm to delete the minimum-priority item (the root) from an n-node heater. What is the expected running time of your algorithm?

3. Suppose we are given a two-dimensional array $M[1..n, 1..n]$ in which every row and every column is sorted in increasing order and no two elements are equal.

(a) Describe and analyze an algorithm to solve the following problem in $O(n)$ time: Given indices i, j, i', j' as input, compute the number of elements of M smaller than $M[i, j]$ and larger than $M[i', j']$.

(b) Describe and analyze an algorithm to solve the following problem in $O(n)$ time: Given indices i, j, i', j' as input, return an element of M chosen uniformly at random from the elements smaller than $M[i, j]$ and larger than $M[i', j']$. Assume the requested range is always non-empty.

(c) Describe and analyze a randomized algorithm to compute the median element of M in $O(n \log n)$ expected time.

Assume you have access to a subroutine Random(k) that returns an integer chosen independently and uniformly at random from the set $\{1 2, \ldots, k\}$, given an arbitrary positive integer k as input.

New CS 473 Spring 2015 — Homework 5 Problem 1

Name:	NetID:
Name:	NetID:
Name:	NetID:

(a) What is the *exact* expected number of flips in one of Rosencrantz's turns?

(b) Suppose Rosencrantz happens to flip k heads in a row on his turn. What is the *exact* expected number of flips in Guildenstern's next turn?

(c) What is the *exact* expected total number of flips (by both Rosencrantz and Guildenstern) in a single game?

New CS 473 Spring 2015 — Homework 5 Problem 2

Name:	NetID:
Name:	NetID:
Name:	NetID:

(a) What is the *exact* expected depth of node j in an n-node heater?

(b) Describe and analyze an algorithm to insert a new item into a heater. Express the expected running time of the algorithm in terms of the priority rank of the newly inserted item.

(c) Describe an algorithm to delete the minimum-priority item (the root) from an n-node heater. What is the expected running time of your algorithm?

New CS 473 Spring 2015 — Homework 5 Problem 3

Name:	NetID:
Name:	NetID:
Name:	NetID:

(a) Describe and analyze an algorithm to solve the following problem in $O(n)$ time: Given indices i, j, i', j' as input, compute the number of elements of M smaller than $M[i, j]$ and larger than $M[i', j']$.

(b) Describe and analyze an algorithm to solve the following problem in $O(n)$ time: Given indices i, j, i', j' as input, return an element of M chosen uniformly at random from the elements smaller than $M[i, j]$ and larger than $M[i', j']$. Assume the requested range is always non-empty.

(c) Describe and analyze a randomized algorithm to compute the median element of M in $O(n \log n)$ expected time.