Analysis of Algorithms-Homework4

Due Tuesday， March 22,2018,11:59 PM

Submit your HW as follows: ontaining the following:

1. A PDF file that contains all the answers to the individual questions, all pictures, all code, and all code output. This should all be well-organized and attractively laid out. This file will contain all the grading feedback after your work is graded.
2. A ZIP file for all Python programs and input files. Also include any other files that you want considered in this Zip file.

Problems

All the problems are from the Cormen, Leiserson, Rivest, and Stein book.

1. (15 points) Show that if L $\geq$ 2, then every binary tree with L leaves contains a subtree having between L/3 and 2L/3 leaves, inclusive.
2. (15 points) Let us associate a "weight" w(q) $= 2^{-depth(q)}$ with each leaf in a binary tree T. Prove that $\Sigma_q w(q) \leq 1$, where the sum is taken over all leaves q in T.

3. (15 points) Do Problem 5.2-4 on p. 122.

5.2-4

Use indicator random variables to solve the following problem, which is known as the ***hat-check problem***. Each of n customers gives a hat to a hat-check person at a restaurant. The hat-check person gives the hats back to the customers in a random order. What is the expected number of customers who get back their own hat?

4. (15 points) Do Problem 5.2-5 on p. 122.

5.2-5

Let $A[1..n]$ be an array of n distinct numbers. If $i < j$ and $A[i] > A[j]$, then the pair (i, j) is called an ***inversion*** of A. (See Problem 2-4 for more on inversions.) Suppose that the elements of A form a uniform random permutation of $\langle 1, 2, \ldots, n \rangle$. Use indicator random variables to compute the expected number of inversions.

5. (15 points) Do Problem 5.4-1 on p. 142.

5.4-1

How many people must there be in a room before the probability that someone has the same birthday as you do is at least $1/2$? How many people must there be before the probability that at least two people have a birthday on July 4 is greater than $1/2$?

6. (15 points) Do Problem 6.5-9 on p. 166. Please implement your algorithm in Python and submit some output to support your claim of performance.

6.5-9

Give an $O(n \lg k)$-time algorithm to merge k sorted lists into one sorted list, where n is the total number of elements in all the input lists. (*Hint:* Use a min-heap for k-way merging.)

7. (10 points) Do Problem 9.3-1 on p. 223.

9.3-1

In the algorithm SELECT, the input elements are divided into groups of 5. Will the algorithm work in linear time if they are divided into groups of 7? Argue that SELECT does not run in linear time if groups of 3 are used.