CS41 Homework 7

This homework is due at 11:59PM on Sunday, October 25. Write your solution using LaTeX. Submit this homework using **github** as a file called **hw7.tex**. This is a **partnered homework**. It's ok to discuss approaches at a high level with others; however, you should **primarily discuss approaches with your homework partner.**

If you do discuss problems with others, you should not reveal specific details of a solution, nor should you show your written solution to anyone else. The only exception to this rule is work you've done with a lab partner *while in lab.* In this case, note (in your **homework submission poll**) who you've worked with and what parts were solved during lab.

1. **Recurrence Relations.** Solve the following recurrence relation.

$$H(n) = 4H(n/2) + 2n^2, \text{ for all } n > 1 \ ,$$

$$H(1) = 5$$

   Use recursion trees or the substitution method.

2. **Summer camp triathlon.** (Kleinberg and Tardos, 4.6)
   Your friend is working as a camp counselor, and is in charge of organizing activities for a set of junior-high-school-age campers. One of the plans is the following mini-triathlon exercise: each contestant must swim 20 laps of a pool, then bike 10 miles, then run 3 miles. The plan is to send the contestants out in a staggered fashion, via the following rule: the contestants must use the pool one at a time. (In other words, first one contestant swims the 20 laps, gets out, and starts biking. As soon as this person is out of the pool, a second contestant begins swimming the 20 laps; as soon as the second person is out and starts biking, a third contestant begins swimming, and so on.)

   Each contestant has a projected *swimming time* (the expected time it will take them to complete the 20 laps), a projected *biking time* (the expected time it will take them to complete the 10 miles of bicycling), and a projected running time (the expected time it will take them to complete the 3 miles of running). Your friend wants to decide on a *schedule* for the triathalon: an order in which to sequence the starts of the contestants. Let's say that the *completion time* of a schedule is the earliest time at which all contestants will be finished with all three legs of the triathalon, assuming they each spend exactly their projected swimming, biking, and running times on the three parts. (Again, note that participants can bike and run simultaneously, but at most one person can be in the pool at any time.) What's the best order for sending people out, if one wants the whole competition to be over as early as possible? More precisely, give an efficient algorithm that produces a schedule whose completion time is as small as possible.

3. **Database Queries.** (Kleinberg and Tardos, 5.1) You are interested in analyzing some hard-to-obtain data from two separate databases. Each database contains n numerical values (so there are $2n$ values total). You'd like to determine the *median* of this set of $2n$ values, defined as the n-th smallest value.

The only way you can access these values is through *queries* to the databases. In a single query, you can specify a value k to one of the two databases, and the chosen database will return the k-th smallest value it contains. Since queries are expensive, you would like to compute the median using as few queries as possible.

- Design an algorithm that finds the median value using at most $O(\log n)$ queries. Full pseudocode is not necessary, but you must clearly explain how it works, and you must handle all edge cases; e.g., do not assume that n is even.
- Show that your algorithm correctly returns the median.
- Prove that your algorithm uses only $O(\log n)$ queries.