DUE FRI FEB 12, 2016 AT 5P VIA SUBMISSION SITE

H2.

You may collaborate with other students on the homework but you must submit your own individually written solution, identify your collaborators, and acknowledge any external sources that you consult. Do no submit a solution that you cannot explain to me.

PROBLEM 1 *Flip-flop*

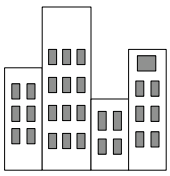
Consider the recurrence $T(n) = 2T(n/2) + f(n)$ in which

$$f(n) = \begin{cases} n^3 & \text{if } \lceil \log(n) \rceil \text{ is even} \\ n^2 & \text{otherwise} \end{cases}$$

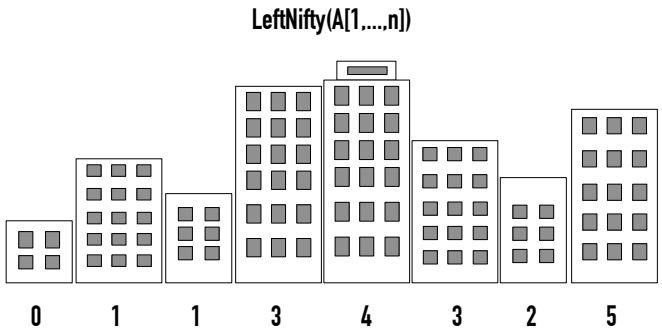
Show that $f(n) = \Omega(n^{\log_b(a)+\epsilon})$. Explain why the third case of the Master's theorem stated above does not apply. Prove a Θ-bound for the recurrence.

PROBLEM 2 *Skyline*

A great skyline must have variations. Define the left skyline function, denoted $n_\ell(s)$, of a skyline s as the total number of times that a building is taller than one of its left neighbors. The right skyline function, $n_r()$, is defined analogously. The skyline s below has 4 buildings, and $n_\ell(s) = 3$ and $n_r(s) = 3$ because building 2 is taller than building 1 and building 4 is taller than buildings 3 and 1, and alternatively, building 1 is taller than 3, and building 2 is taller than 3 and 4.



As another example, this skyline has $n_\ell = 19$, the contribution of each building is listed below the building.



Design and analyze a divide and conquer algorithm that computes the right and left skyline functions of a skyline s with n buildings. The input $A[1, \ldots, n]$ consists of the heights of each building along a street; assume all buildings have unique heights. Your solution should have a running time of $\Theta(n \log n)$.

PROBLEM 3 *Trolley*

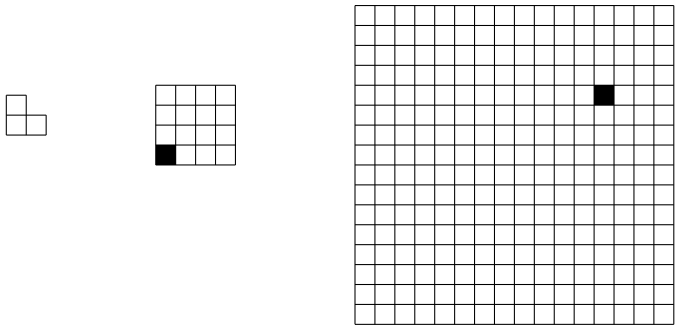
The city of Charlottesville commissions you to design a new bus system for Main Avenue which has n stops on the North-bound route (lets ignore the South-bound route). Commuters may begin their journey at any stop i and end at any other stop $i < j$. There are some obvious options:

1. You can have a bus run from the southern-most point to the northern-most point as a traditional busline might run. The system would be cheap because it only requires n segments for the entire system. However, a person traveling from stop $i = 0$ to stop $j = n$ must travel through all n segments. This system will be slow.
2. You can have a special express bus run from every point to every other destination. No person will every wait through any unnecessary segments no matter where they start and end. However, this system requires $\Theta(n^2)$ segments and will be expensive.

Suggest a compromise solution: Use a divide-and-conquer technique to design a bus system that uses $\Theta(n \log n)$ route segments and which requires a person to wait through at most 1 extra segment when going from any i to any j (as long as $i \leq j$, i.e., we only consider North-bound routes for simplicity, and all buses run North). In other words, a commuter can travel from any i to any j by using at most 2 segments.

PROBLEM 4 *Tiling*

An *L-tile* is an L-shaped tile formed by 1-by-1 adjacent squares. The problem is to cover any 2^b-by-2^b chessboard with one missing square (anywhere on the board) with L-tiles. A solution must cover all squares except the missing one and no two tiles can overlap.



(a) (b) (c)

Figure 1: (a) An L-tile (b) A 4×4 instance of the problem (c) A 16×16 instance

An instance is specified by b (which determines the size of the board), and the coordinates $(x, y) \in [1, \ldots, 2^b] \times [1, \ldots, 2^b]$ of the missing square. A solution to an instance consists of a list of triples, where each triple describes the position of one of the tiles.

1. Describe a divide and conquer strategy for solving this problem. (Hint: Start with the base case.) Brute force search will not receive credit. Provide pseudo-code for your solution. You may use macros such as UPPERRIGHT(A) or LOWERLEFT(A) to refer to the upper-right (lower-left) quadrant of a two-dimensional array A.
2. Show a tight asymptotic bound for the running time of your solution.