

- The assignment is due at Gradescope on September 22 at 11:59pm. Late assignments will not be accepted. Submit early and often.
- You are permitted to study with friends and discuss the problems; however, *you must write up your own solutions, in your own words*. Do not submit anything you cannot explain. If you do collaborate with any of the other students on any problem, please do list all your collaborators in your submission for each problem.
- Finding solutions to homework problems on the web, or by asking students not enrolled in the class is strictly prohibited.
- We require that all homework submissions are prepared in Latex. If you need to draw any diagrams, however, you may draw them with your hand. Please use *exactly 1 page* for your answers.

PROBLEM 1 *Asymptotic Notation Review*

Rank the following functions by order of growth; that is, find an ordering f_1, f_2, \dots, f_{12} of the functions satisfying $f_i = O(f_{i+1}) \forall i \in \{1, \dots, 11\}$. You do not need to provide justification. Hint: use logarithms to simplify the functions.

$$\begin{array}{ccccccc}
 n^3 & 7^{\log_4 n} & n! & 2^{(\log_2 n)^2} & \log_2(n!) & n^{1/(\log_2 n)} \\
 \log_2 \log_2 n & (\log_2 n)^{(\log_2 n)/(\log_2 \log_2 n)} & \sqrt{n} & 2^{\log_3 n} & 3^{\sqrt{\log_2 n}} & 1.01^n
 \end{array}$$

PROBLEM 2 *Karatsuba Example*

Carry out the Karatsuba algorithm for $1234 \cdot 9876$.

PROBLEM 3 *Divide and Conquer*

The Greek Diogenes spent his life searching for an honest man. Consider the Greek system today with a group of n sorority sisters and fraternity brothers. We can test a pair—lets call them Alice and Bob for simplicity—by asking them whether the other is honest. Honest Greeks always report truthfully, but the dishonest can report arbitrarily. Thus, the following outcomes are possible:

Alice says	Bob says	
"Bob is honest"	"Alice is honest"	Either both are honest or both are dishonest
"Bob is honest"	"Alice is dishonest"	at least one is dishonest
"Bob is dishonest"	"Alice is honest"	at least one is dishonest
"Bob is dishonest"	"Alice is dishonest"	at least one is dishonest

1. A group of n Greeks is *moral* if more than half are honest.

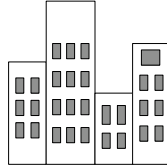
Suppose we start with a moral group of n Greeks. Describe a method that uses only $\lfloor n/2 \rfloor$ pair-wise tests between the Greeks to find a smaller moral group of at most $\lceil n/2 \rceil$ Greeks.

Using this approach, devise an algorithm that classifies all Greeks as honest or dishonest using only $\Theta(n)$ pairwise tests. Prove the correctness of your algorithm and prove that only $\Theta(n)$ tests are used.

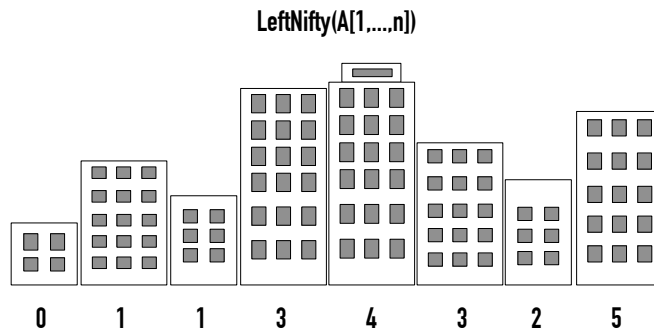
- 2*. Society may fall apart if there is not an honest majority. Prove that a conspiracy of $\lfloor n/2 \rfloor + 1$ dishonest Greeks (who may, for example, know each other and share a plan) can foil any attempt to find an honest person. To solve this problem, you must rule out *any* algorithm, not just the one you proposed above.

PROBLEM 4 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, $n_r(\cdot)$, 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 $A[1, \dots, n]$ consists of the heights of each building along a street in left to right order; assume all buildings have unique heights. Your solution should have a running time of $\Theta(n \log n)$.

PROBLEM 5 *Programming Skyline*

In this part, you will implement your algorithm for Skyline and test it on real data. Register and take on the challenge at <https://www.hackerrank.com/contests/cs4800f16>

PROBLEM 6 *Trolley*

The city of Boston commissions you to design a new bus system for Huntington 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 for that person.
2. You can have a special express bus run from every point to every other destination. No person will ever 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.