

Problem Set 3

DATA STRUCTURES AND ALGORITHMS 2 - FALL 2026

mst3k

due september 30, 2026 at 11:59p

Collaboration Policy: You are encouraged to collaborate with up to 4 other students, but all work submitted must be your own *independently* written solution. List the computing ids of all of your collaborators in the `collabs` command at the top of the tex file. Do not share written notes, documents (including Google docs, Overleaf docs, discussion notes, PDFs), or code. Do not seek published or online solutions for any assignments. If you use any published or online resources (which may not include solutions) when completing this assignment, be sure to cite them. Do not submit a solution that you are unable to explain orally to a member of the course staff. Any solutions that share similar text/code will be considered in breach of this policy. Please refer to the syllabus for a complete description of the collaboration policy.

Collaborators: list your collaborators

Sources: list your sources

PROBLEM 1 *Solving Recurrences*

Prove a (as tight as possible) O (big-Oh) asymptotic bound on the following recurrences. You may use any base cases you'd like. It may be helpful to draw out the tree; however, you should prove the asymptotic bound using induction.

- $T(n) = T(\frac{n}{2}) + T(\frac{n}{4}) + T(\frac{n}{8}) + n$

Solution:

- $T(n) = 2T(\frac{n}{3}) + T(\frac{n}{6}) + n$

Solution:

PROBLEM 2 *Receding Airlines*

You have been hired to plan the flights for Professor Hott's brand new passenger air company, "Receding Airlines." Your objective is to provide service to n major cities within North America. The catch is that this airline will only fly you East (and, of course, it can't travel all the way around the world to get back west, as that would take too much fuel).

You recognize that in order to enable all your passengers to travel from any city to any other city (to the East) with a single flight requires $\Omega(n^2)$ different routes. Prof. Bloomfield says that the airline cannot be profitable when supporting so many routes. Another option would be to order the cities in a list (from West to East), and have flights that go from the city at index i , to the city at index $i + 1$. This requires $\Theta(n)$ routes, but would mean that some passengers would require $\Omega(n)$ connections to get to their destination.

1. Devise a compromise set of routes which requires no passenger have more than a single connection (i.e. must take at most two flights), and requires no more than $O(n \log n)$ routes. Prove that your set of routes satisfies these requirements; that is, write out the recurrence relation and prove it is in $O(n \log n)$.

Solution:

2. After a few years, passengers start demanding routes from East to West, and you decide to support new routes from East to West (in addition to supporting routes from West to East). Show that with routes in *both* directions, it is possible to connect all n cities with just $O(n)$ routes such that no passenger needs more than a single connection to get to their destination.

Solution:

Note: In class so far, we've used recurrence relations to count an algorithm's time complexity, i.e., how many basic operations are executed. Here we're using one to count the number of flights. While this is a bit different than measuring an algorithm's time-complexity, we can still use a divide and conquer approach and a recurrence relation to answer these questions.

PROBLEM 3 *Mission Impossible*

As the newly-appointed Secretary of the Impossible Missions Force (IMF), you have been tasked with identifying the double agents that have infiltrated your ranks. There are currently n agents in your organization, and luckily, you know that the majority of them (i.e., strictly more than $n/2$ agents) are loyal to the IMF. All of your agents know who is loyal and who is a double agent. Your plan for identifying the double agents is to pair them up and ask each agent to identify whether the other is a double agent or not. Agents loyal to your organization will always answer honestly while double agents can answer arbitrarily. The list of potential responses are listed below:

Agent 001	Agent 002	Implication
"002 is a double agent"	"001 is a double agent"	At least one is a double agent
"002 is a double agent"	"001 is loyal"	At least one is a double agent
"002 is loyal"	"001 is a double agent"	At least one is a double agent
"002 is loyal"	"001 is loyal"	Both are loyal or both are double agents

1. A group of n agents is "acceptable" for a mission if a majority of them ($> n/2$) are loyal. Suppose we have an "acceptable" group of n agents. Describe an algorithm that has the following properties:
 - Uses at most $\lfloor n/2 \rfloor$ pairwise tests between agents.
 - Outputs a smaller "acceptable" group of agents of size at most $\lceil n/2 \rceil$.

Solution:

2. Using your approach from Part 1, devise an algorithm that identifies which agents are loyal and which are double agents using $\Theta(n)$ pairwise tests.

Solution: