Analysis of Algorithms
Lab 4

Andrés Calderón, Ph.D.

February 27, 2025

1 Introduction

In this lab, we will continue our exploration of algorithm analysis by tackling a new set of problems from MIT's OpenCourseWare Design and Analysis of Algorithms (6.046J) course. This lab focuses on key topics such as asymptotic growth, recurrence relations, and an applied problem in computational geometry.

The goal of this lab is not just to find answers but to deepen your understanding of algorithmic complexity and problem-solving strategies. You will begin by analyzing the growth rates of various functions and solving recurrence relations to determine tight asymptotic bounds. These exercises will strengthen your ability to assess algorithm efficiency—a fundamental skill in computer science.

Following these theoretical exercises, you will engage with a practical problem related to radio frequency assignment. This problem will challenge you to design an efficient algorithm that ensures radio stations can operate without interference, incorporating techniques such as grid-based partitioning and divide-and-conquer strategies.

Additionally, a key component of this lab is a comparative study of AI-driven problem-solving. After working through the problems individually and collaboratively, you will evaluate the effectiveness of AI frameworks in generating solutions, comparing their approaches with your own. This exercise will help you critically assess AI-generated outputs and refine your problem-solving intuition.

By the end of this lab, you will have gained deeper insights into algorithmic analysis, collaborative problem-solving, and the evolving role of AI in computational research.

This problem set comes from the *Design and Analysis of Algorithms (6.046J)* course. The problem set is available here.

Let's get started!

2 Asymptotic Growth

Sort all the functions below in increasing order of asymptotic (big-O) growth. If some have the same asymptotic growth, then be sure to indicate that. As usual, lg means base 2.

1. $5n$

2. $4 \lg n$

3. $4 \lg \lg n$

4. n^4

5. $n^{\frac{1}{2}} \lg^4 n$

6. $(\lg n)^{5 \lg n}$

7. $n^{\lg n}$

8. 5^n

9. 4^{n^4}

10. 4^{4^n}

11. 5^{5^n}

12. 5^{5n}

13. $n^{n^{\frac{1}{5}}}$

14. $n^{\frac{n}{4}}$

15. $\left(\frac{n}{4}\right)^{\frac{n}{4}}$

3   Solving Recurrences

Give asymptotic upper and lower bounds for $T(n)$ in each of the following recurrences. Assume that $T(n)$ is constant for $n \leq 2$. Make your bounds as tight as possible, and justify your answers.

a) $T(n) = 4T\left(\frac{n}{4}\right) + 5n$

b) $T(n) = 4T\left(\frac{n}{5}\right) + 5n$

c) $T(n) = 5T\left(\frac{n}{4}\right) + 4n$

d) $T(n) = 25T\left(\frac{n}{5}\right) + n^2$

e) $T(n) = 4T\left(\frac{n}{5}\right) + \lg^5 n \sqrt{n}$

f) $T(n) = 4T\left(\frac{n}{5}\right) + 5n$

g) $T(n) = 4T(\sqrt{n}) + \lg^5 n$

h) $T(n) = 4T(\sqrt{n}) + \lg^2 n$

i) $T(n) = T(\sqrt{n}) + 5$

j) $T(n) = T\left(\frac{n}{2}\right) + 2T\left(\frac{n}{5}\right) + T\left(\frac{n}{10}\right) + 4n$

4   Radio Frequency Assignment

Prof. Wheeler at the Federal Communications Commission (FCC) has a huge pile of requests from radio stations in the Continental U.S. to transmit on radio frequency 88.1 FM. The FCC is happy to grant all the requests, provided that no two of the requesting locations are within Euclidean distance 1 of each other (distance 1 might mean, say, 20 miles). However, if any are within distance 1, Prof. Wheeler will get annoyed and reject the entire set of requests.

Suppose that each request for frequency 88.1 FM consists of some identifying information plus (x, y) coordinates of the station location. Assume that no two requests have the same x coordinate, and likewise no two have the same y coordinate. The input includes two sorted lists, L_x of the requests sorted by x coordinate and L_y of the requests sorted by y coordinate.

(a) Suppose that the map is divided into a square grid, where each square has dimensions $\frac{1}{2} \times \frac{1}{2}$. Why must the FCC reject the set of requests if two requests are in, or on the boundary of, the same square?

(b) Design an efficient algorithm for the FCC to determine whether the pile of requests contains two that are within Euclidean distance 1 of each other; if so, the algorithm should also return an example pair. For full credit, your algorithm should run in $O(n \lg n)$ time, where n is the number of requests.

> **Note**
>
> It could be helpful to have a look at the lecture from Prof. Srinivas Devadas from the course available at YouTube. Part of this lecture is about how to solve **Convex Hull** using the *Divide & Conquer* paradigm. Focus only on minute 7:00 until 23:00.

5   What We Expect

We already know that the answers for this problem set are available on the course webpage. For sections 2 and 3, we expect you to make an honest attempt at solving the exercises on your own. If you consult the solutions, it should only be for reference.

However, for the assignment in section 4, we will try a different approach this time. First, we ask you to spend an hour working individually on the problem—just by yourself. No Internet, textbooks, or solutions are allowed during this phase; only you, your notes, a pen, and a piece of paper. Then, you will discuss your findings with your group, talk about each of your approaches, and identify their pros and cons. Spend another hour or so on this discussion, and at the end, conclude with your proposed solution. Again, no Internet, textbooks, or solutions up to this point.

Next, we want you to ask a couple of AI frameworks for a solution. For example, ask ChatGPT what the solution would be for the assignment (just section 4). Ask as many questions as you need for clarification, and make sure to request elaboration on every detail. Capture all the prompts from your interaction. Then, you will ask another AI framework, such as DeepSeek. Ask the same questions and request all the elaborations, and don't forget to capture all your prompts.

Now it is time to analyze your findings. How does one AI framework differ from the other? How do both of them compare to your solution at each step of the proposal? Do you feel you performed well? We want to know **ALL** about this analysis. Do not hesitate to be as detailed as possible. Write out all your findings and formulate your own conclusions.

Finally, at this point, go and check the ground truth (a.k.a. the solution). What do you think about it? Elaborate on your answer and be specific.

You will compile all your answers into a well-structured report and submit it before the lab on **March 13th, 2025**.

Note: Sections 2 and 3 must be handwritten. Please scan your report in PDF format and submit it to my email with the `[ADA]` tag + lab number in the subject line.

Happy Hacking ☺!