Introduction to Algorithmic Problem Solving

 (https://github.com/learn-co-curriculum/phase-1-algorithms-intro-to-algorithmic-problem-solving) (https://github.com/learn-co-curriculum/phase-1-algorithms-intro-to-algorithmic-problem-solving/issues/new)

Learning Goals

- Understand what you will learn in this section

Introduction

Welcome to this section on algorithmic problem solving! From Software Engineering Prep up to this point, you've learned the fundamentals of JavaScript as a language, including how to write functions and work with built-in data types, as well as how to use the "three pillars" of JavaScript (DOM Manipulation, Event Handling, and Network Communication) to build small web applications.

In this section, we're going to shift gears a bit and leave the browser to focus on one of the most important skills you can build as a software engineer: developing a **process** for solving problems with code.

Developing your algorithmic problem solving skills will help you approach labs, code challenges and projects throughout the rest of this program. Just as importantly, it will also help immensely as you begin to prepare for technical interviews once you graduate the program! One of the most critical steps toward landing a job after graduating is being able to demonstrate your proficiency writing code during technical interviews, and having a rock-solid problem solving process in place will give you much more confidence when confronted with a new problem.

The objectives for this section are to:

- Define what "algorithm" means
- Develop a process for algorithmic problem solving
- Understand the usefulness of Big O notation
- Identify the Big O time and space complexity of a given algorithm
- Recognize common Big O runtimes: constant, logarithmic, linear, and quadratic

By the end of this section, you'll have a strong foundation in the skills needed to solve many data structures and algorithm problems. Later in the program, we'll revisit this material and move on to more advanced topics.

Resources

Below are some popular sites for practicing data structures and algorithms. We recommend bookmarking these and returning to them later for practice once you've worked through this section.

- **Leetcode** ↪ **(https://leetcode.com/)**
- **HackerRank** ↪ **(https://www.hackerrank.com/)**
- **codewars** ↪ **(https://www.codewars.com/)**