General Assembly

STUDY GUIDE

SEARCH ALGORITHMS

Key Terms

- **Brute force:** Refers to an algorithm that tries every possibility available.
- **Binary search:** An algorithm that finds the position of a target value within a sorted array by dividing the array in half until the value is found. Can be written with a loop or with recursion.

Cheat Sheet

Big O review

- Brute force search has a time complexity of **O(N)** because it looks at every element in the array.
- Binary search has a time complexity of **O(log(N))** because it divides the array in half in each iteration.

How binary search works

When searching for a given value in a sorted array, the binary search algorithm will...
1. Find the middle element in an array.
2. Is this the value? If so, we're done! If not, keep looking.
3. If the value should come before the middle element, search the subsection of the array from 0 to the element just before the middle.
4. If the value should come after the middle element, search the subsection of the array from just after the middle element to the end.
5. Repeat the process of finding the middle element of the subsections and dividing the sections in half until you find the element.
6. If you get down to a 0- or 1-element array and still haven't found the value, you know it's not present.

- If the value is in the array, binary search will return its index.
- If the value is not in the array, binary search will return -1.
- If the value isn't present, you can also adapt the algorithm to return the index where the value *should* appear. This value will always be a negative number. This adaptation is helpful if you're trying to add missing values to the array.