MAKE SCHOOL

Algorithm Analysis

Check In

Share your pets!! 😸🐶🐰🐍🐦

What We’re Going to Learn

- What is time complexity and how do we write it
- Why use Big O notation
- Some common time complexities
- How to analyze a piece of code in terms of time complexity

Interview Question Warmup

Write a method that will clone a Linked List

Students browse: repl.it/@MakeSchool/linkedlistinterviewwarmup?lite=true

Pear Deck Interactive Slide
Do not remove this bar

Motivation

We’ve studied two data structures so far:

Arrays

Linked Lists

MAKE SCHOOL

Compare and Contrast Arrays and Linked Lists

Students, write your response!

Pear Deck Interactive Slide
Do not remove this bar

Why are there so many different types of data structures?

Students, write your response!

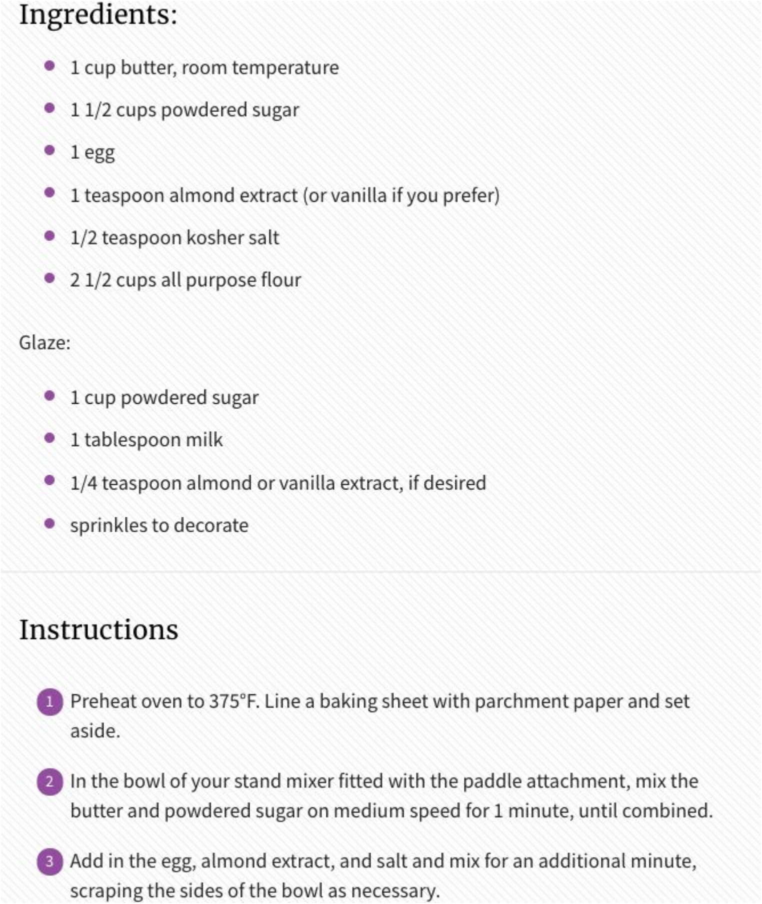
Pear Deck Interactive Slide
Do not remove this bar

MAKE SCHOOL

What is an algorithm?

Students, write your response!

An Algorithm is like a recipe: a series of well defined steps to solve a problem



Ingredients:

- 1 cup butter, room temperature
- 1 1/2 cups powdered sugar
- 1 egg
- 1 teaspoon almond extract (or vanilla if you prefer)
- 1/2 teaspoon kosher salt
- 2 1/2 cups all purpose flour

Glaze:

- 1 cup powdered sugar
- 1 tablespoon milk
- 1/4 teaspoon almond or vanilla extract, if desired
- sprinkles to decorate

Instructions

1. Preheat oven to 375°F. Line a baking sheet with parchment paper and set aside.
2. In the bowl of your stand mixer fitted with the paddle attachment, mix the butter and powdered sugar on medium speed for 1 minute, until combined.
3. Add in the egg, almond extract, and salt and mix for an additional minute, scraping the sides of the bowl as necessary.

Think about this

MAKE SCHOOL

I want to time how long it takes to insert 10,000 items into both an array and a linked list

- What if I timed them on my computer?
- What if my friend timed them on their computer?
- Would we get the same result?

Students, write your response!

Time Complexity Analysis of Algorithms

Time complexity analysis gives us a formalized way to compare algorithms without being reliant on specific configurations and hardware

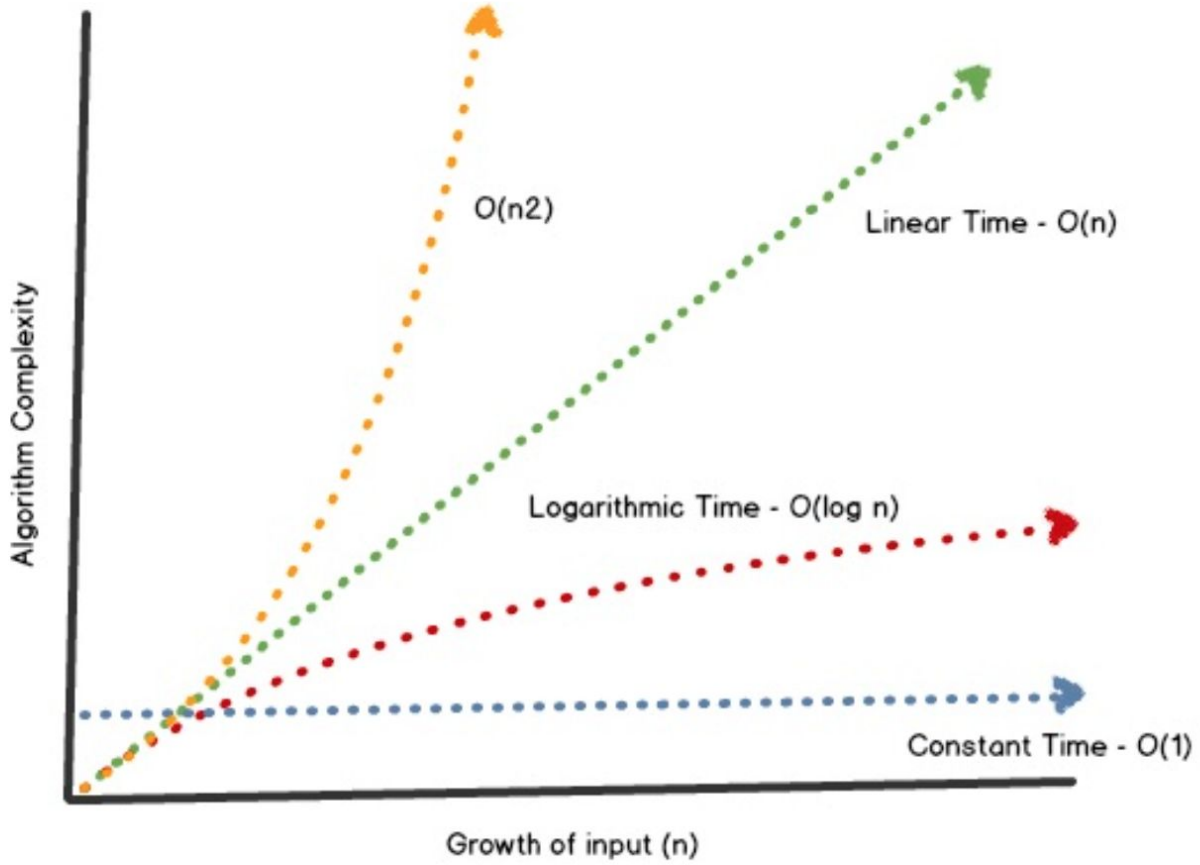
To analyze the time complexity of an algorithm we analyze how the time changes when the input size grows

Input size is usually denoted the letter n

Being able to analyze algorithms in terms of time complexity is key to understanding how different data structures work

(also you will absolutely need to know this for technical interviews)

Time Complexity Analysis



Big O Notation

- O(1): fast
- O(n): pretty fast
- O(n^2): pretty slow

Big O Notation

- O(1): constant time
- O(n): linear time, increases with the input n
- O(n^2): Quadratic time, increases exponentially with input n

Let’s look at some examples!

Best Case and Worst Case

We can think about time complexity with two different lenses

-Best Case: If conditions are ideal how fast could this run

- Worst Case: The worst possible conditions that would make our algorithm run the slowest'

Which one do you think is more important?

Students, write your response!

Pear Deck Interactive Slide
Do not remove this bar

What about Linked List and arrays? Let’s walk through analyzing prepend()!

Big O Linked List Worksheet

Shout Outs