WEEKLY NEWSLETTER

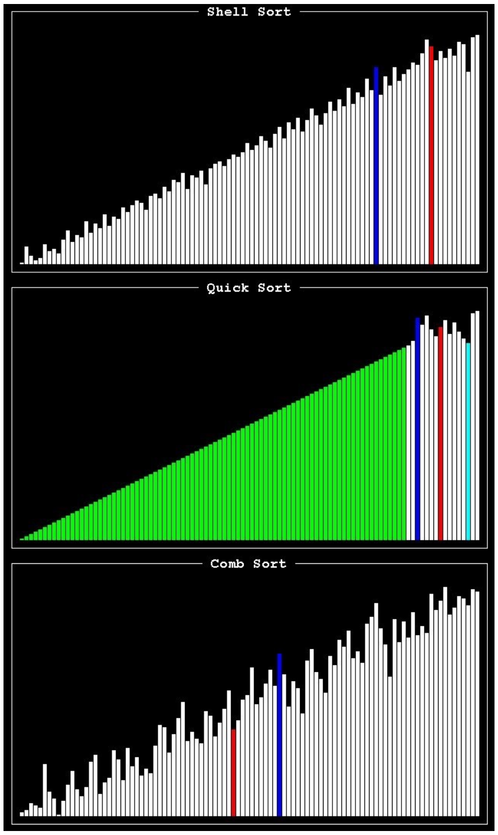
Sort!

Beauty of orders.

As a young computer scientist, ICS4U is/was/is going to be one of your best memories in high school. Graphics, I/Os, objects make the course curriculum quite relevant and practical. However, there is only one thing, that you teacher teaches, you don't quite get what's for. It's like your advanced functions homework, which sometimes makes you wonder: will I ever have to draw these wavy graphs again? The thing mentioned above, is "sorting".

> **"Don't write them. Use a library."**
>
> *-AN EXPERIENCED PROGRAMMER*

Why is sorting so important that every mature computer scientist has encountered somewhere in their lives? It is because, review your life, if you don't know how to efficiently sort an array, Google would never hire you. Now reflect again, when you finally become a manager at Google, without using the sorting function in Google Spreadsheets, how long would it take you sort your hundreds of stuffs' information by last names in alphabetical order? Well, I get your point, sorting hundreds of names is handled already, why are we still studying so many different kinds of sorting methods (which will be introduced in a moment)? It is because everyone has OCD. We have a natural tendency to separate m&m's into their own colours whenever we see them mixed together. Or, simply because you want to be hired by Google. That's all.



A popular YouTube video visualized different sorting methods. Watch it at: https://youtu.be/ZZuD6iUe3Pc

Now with the knowledge of why should you study sorting methods, you are ready for a general introduction. A sorting algorithm's goal, is to put a random collection of numbers or other sortable things into ascending or descending order. There are a lot of sorting methods, which are all fighting toward the same goals: *quicker, smaller, stabler*.

Quicker

The manner of not wasting anybody's time is not only appreciated among humans, but also among sorting methods. Sort methods, like *Heap Sort* and *Quick Sort,* with its modest name, are very well known due to their impressive speed; hence you are going to study their convoluted algorithms at some point in your Computer Science 101. While *Bubble Sort* and *Insertion Sort* are also impressive because of their simplicity to understand, not their mediocre speed.

Smaller

Heap Sort takes the advantage of memory space in order to be acclaimed as "one of the fastest sorting methods". No sane man wants to fill up his/her beloved computer's RAM and crash the precious yet fragile program he/she has been debugging on overnight, fuelled by 20 cups of coffee. Therefore, taking less space is valued just as much as being fast, if not more.

Stabler (Optional)

What does it mean for a sorting algorithm to be stable? Good question. When it comes to real life applications, there are usually other attributes attached to the entries you are sorting. Being stable means after you sort your array, the elements that are treated to be the same in your sorting condition, however still distinguishable due to other attributes, still follow their original order. For (a real-life) example, you are trying to sort your entire math class by their heights. Before sorting, Alice was standing before Bob, who is just as tall as Alice; after your sorted everyone according to heights, if Alice is still standing in front of Bob, then your sorting method is stable, and vice versa. In this example, Alice and Bob can be distinguished because of their unique appearances and names. They are treated the same because the sorting is based on their heights. According to the original order, Alice was in front of Bob; hence in a stable method, Alice should be still standing in front of him despite the fact they are treated the same according to sorting condition.

It is not a hard thing to change an unstable method into a stable one. By giving a card that has the original order of each student to the class, then compare Alice's and Bob's original-order cards after sorting, you can easily judge who get to be standing in front of the other. It is called "key comparison" in computer science.

Furthermore, the three parameters are valued differently under different circumstances. The same goes to different sorting methods. Programmers have been optimizing a variety of sorting algorithms to sort integers, decimals, strings and so on. Hence, there isn't an absolute ruling method of all.

Weekly Programming Challenge

Hired - Role Play

Time: 10 min prep, 2 min demo

Team: 2 people

Background: You have just graduated from University Of California, Berkeley, and co-founded a startup called "Sortable" in Silicon Valley. At the meanwhile, Google has been struggling to create an innovative sorting method to implement to their cloud Spreadsheet, so the company decided to acquire your young and potential startup, since your company has "revolutionizing sorting solutions" as its mission. Your first meeting with the company, is on Wednesday, October 29th, 2015.

Task: In 10 minutes, you and your co-founder have to come up with a sorting method before the meeting and draw a flowchart on a piece of paper to demonstrate your method effectively. Your team has to pitch your sorting method to Google (you only need to present. It will be a safe school environment) during the meeting within 2 mins.

How to draw an algorithmic flowchart:

https://www.edrawsoft.com/How-to-draw-flowchart.php

Tip: You can Google the sorting methods up mentioned in this week's newsletter and get inspired. Or come up with your original. Remember you only have 2 minutes to present.

> **"Until then, keeping coding!"**
>
> *—THIS NEWSLETTER*