Lab 1: Heap in JavaScript

Due Date: Before Lab on Week 2

Objectives

This lab should be a refresher on how to program. It's been a hell of a year and a heck of a summer break. Because I mean business, I'm going to ask you to write a slightly harder data structure that a queue or stack. You only need the slides from Lecture 1 to be able to complete this lab.

By the end of the lab, you should be:

1. Familiar with basic Javascript types like arrays and ints
2. Able to execute a node.js file

Grading

1. 90 points: Code functionality
2. 10 points: Code Styling & Format
3. -20 points: Deliverable doesn't match description below

Deliverable

Your lab should be a single zip with the following naming scheme: student_id-gmail_id.zip. The only 3 files inside this zip are

1. heap.js file
2. heap.test.js file
3. screenshot.png (or jpg)

Task

The goal of this lab is to implement a working min-heap in JavaScript using an array. Write the three functions:

1. getMin() - returns the value of the root node from the min-heap (without popping)
2. insert() - adds a new node into the heap while keeping the heap property of parents being smaller than child nodes

3. popMin() - removes (and returns) the root node from the min heap and rebalances the heap

Use the given heap.js file as the starting point for your work. It has some helper functions for you already.

As you're coding, try testing out your functions using the heap.test.js file. This file itself won't be graded so you can feel free to modify and play around with it. Note that the test cases in the testHeap file are labelled with the points you get for each test case.