lab 16 Binary Heap and Heapsort

Instructions: This lab is a practice in constructing a basic binary heap and performing heap sort. Implement insert, copy constructor, getHeight, getSize, contains, removeFirst, operator[], and sort. Note: please follow the lecture as some of this will be done in class.

```
#ifndef HEAP_H
#define HEAP_H

#include <string>

template<class T>
class Heap {
    private:
        /* Lets fill out in class. */
    public:

        /* Creates an empty heap. */
        Heap();

        /* Does a deep copy of the array into the heap. */
        Heap(const T *array, const int size);

        /* Add a given value to the heap.
         * Must maintain ordering!
         */
        void insert(const T &val);

        /* Returns the height of the heap. */
        int getHeight();

        /* Returns the size of the heap. */
        int getSize();

        /* Returns the index if an item if exists in the heap.
         * Otherwise return -1.
         */
        int contains(const T &val) const;

        /* Retrieves the element at position pos */
        T& operator[](const int pos);

        /* Removes and returns the first element */
        T& removeFirst();

```

```
40          /* Performs a Heap Sort and returns an array of the sorted
41           * elements.
42           * Note: the heap is empy after the sort!
43           */
44          T* heapSort();
45
46          ~Heap();
47  };
48
49  /* Since heap templated, we include the .cpp.
50   * Templated classes are not implemented until utilized (or explicitly
51   * declared.)
52   */
53  #include "heap.cpp"
54
55  #endif
```

Write some test cases:
Create some test cases, using Unity, that you believe would cover all aspects of your code.

Memory Management:
Now that are using new, we must ensure that there is a corresponding delete to free the memory. Ensure there are no memory leaks in your code! Please run Valgrind on your tests to ensure no memory leaks!

How to turn in:
Turn in via GitHub. Ensure the file(s) are in your directory and then:

- $ git add <files>
- $ git commit
- $ git push

Due Date: October 19, 2020 2359

Teamwork: No teamwork, your work must be your own.