**Introduction**

For this lab we are given certain part of the Min-Heap Implementation, we have to complete the rest of the implementation. Furthermore, we are going to use the “UnitTest” library to make sure that the heap is producing the correct outputs. Finally, once the min-heap implementation is finish we are going to use it to create the heap-sort method. We are going to use this method and the heap to sort the numbers that are in a separate text file under the same directory. The final result should be the numbers in the text file sorted in the console where we are going to print the result.

**Proposed Solution**

I approached this problem by first drawing the tree and creating the methods that are going to retrieve the parent and the children of the given index. I am going to use these methods for reordering the nodes of the heap after an insertion or an extraction of the min element. There are going to be two methods that are going to reorder the heap; one method is going to reorder the heap from the bottom up and the other one is going to reorder it from the root to the leaf, each one following a specific path. I am going to need two different ways of reordering because in inserting you insert the element in the next available leaf, that means that you have to compare the node to its parents all the way to the root. If one parent is bigger that the node, then I need to swap both nodes, this is to keep the heap from going smallest to largest. However, if extract the min node from the root and replace it with the last leaf, then I have to compare the root to the children; swapping the nodes that violate the rule. After I completed these two methods, I then completed the insert method by appending the given node to the list and reordering the heap from the bottom to the top. Finally, the last method for the heap class was the extract-min method. This method returns the root, replaces the root with the last leaf, and makes the heap shorter by one index. I use the reordering method that operates from the root to the leaf nodes; once again swapping the nodes that violate the rule. Now that the heap class is done, I implemented the heap-sort method. This method should insert all the numbers to a heap and extract them; once extracted, they are going to be sorted. The final part of the project was just reading the numbers from a text file and giving them to the heap-sort method.

**Experimental Results**