**Cary Anderson**

CS 241 - Project 2

Professor Tang

Binary Heap

Date: 3 May, 2014

Due Date: 9 May, 2014

**Section 1: Project Description:**

Build a BinaryHeap class and a program to test the efficiency of building a heap given 100 randomly generated integers. Allow for user to select from two options: 1.) give the average number of swaps (over 20 sets) for the insertion method of building a binary heap and also the optimal method 2.) output the first 10 integers (out of 100) that are being used to build the binary heap, output the number of swaps for both methods, perform 10 removals and then print the first 10 values in the finished array.

**Section 2: Project Specification:**

Output averages over 20 sets of 100 randomly generated integers (ranging from 1 – 1000). Use series of insertions (heap.addNode(i)) to build the heap and then compare with

***BinaryHeap heap = new heap(array)***

**Section 3: Testing methodology:**

To test the program, I originally just tested by building heaps and using a method that I named BinaryHeap.print() which would just display the contents of the binary heap array to ensure that the private BinaryHeap.heapify() methods were working properly. Then after ensuring that they were able to sort effectively I implemented the test specifications by taking the average number of swaps over 20 sets of 100 integers.

**Section 4: Lessons Learned:**

I learned how to build a complete tree using an array. This project seemed pretty straight forward.

**Section 5: Output:**

So in my overall results my average number of swaps for the insertion method were about 110.1 and my average number of swaps for the optimal method were about 96.3.