**Data Structures (2080C) – Lab 9**

***Topics covered: Heaps and Priority Queues***

**Objective:**

The objective of this Lab is to create an implementation of a Heap and a Priority Queue.

**Task 1:**

1. Create an implementation of a Priory Queue using a Queue. Include the following members:
   1. Insert – This will add an item to the queue in the appropriate location.
   2. Remove – This will remove an item from the queue from the appropriate location.
   3. PrintQueue – This will print the items in the queue in priority order.
2. Create unit Tests for testing the queue.
3. In the lab report, discuss the advantages and disadvantages of this structure over a standard queue.

**Task 2:**

1. Create an implementation of PriorityQueue using a Heap. Include the following members:
   1. Insert – This will add an item to the queue in the appropriate location.
   2. Remove – This will remove an item from the queue from the appropriate location.
   3. PrintHeap – This will print the items in the heap in priority order.
2. Create a unit Tests for testing the Heap.
3. In the lab report, discuss the advantages and disadvantages of this structure over a standard binary search tree.

**Task 3:**

1. Evaluate the performance of adding 500, 1,000, 2,000, and 5,000 random items to each of the ADT built in this using the same techniques as the sorting lab. Make sure the items being added are the same for each of the structures. In the lab report, graph the results and discuss which structure performs which.
2. Evaluate the performance of removing all of the items from step 1 for each of the 4 sets of data (remove 1 at a time, not all at once). Make sure the items being added are the same for each of the structures. In the lab report, graph the results and discuss which structure performs which.

**Lab Submission:**

1. Write a lab report including the following information:
   1. A description of the objectives/concepts explored in this assignment including why you think they are important to this course and a career in Engineering.
   2. A demonstration of the output of your testing.
   3. Requested evaluations from task descriptions
   4. Chart of work done by members. If you have a deadbeat group member this is your opportunity to tell me. If you all worked equally, then the percentages will be roughly 100/N where N is number of group members.
2. Include all source code from all tasks, input and output files (if any), and any special instructions to compile and run those programs.
3. Package all files in a single zip folder and submit the file as a group via Blackboard. Make sure all group member’s names are on the lab report.

**Lab Grading:**

1. 20% - Lab attendance
2. 20% - Task 1 has been correctly implemented.
3. 20% - Task 2 has been correctly implemented.
4. 20% - Task 3 has been correctly implemented.
5. 20% - Lab report contains all required information and is well written.

If program fails to compile, 0% will be given for that Task.