**Data Structures (2028C) -- Spring 2017 – Lab 12**

***Topics covered: Sorting***

*Lab due:* ***Sunday, Apr 15 at 11:55PM for Monday Section***

***Tuesday, Apr 17 at 11:55 PM for Wednesday Section***

**Objective:**

The objective of this homework is to investigate different sorting algorithms.

**Task 1:** Create a program that generate an array of sizes n= 10, 100, 500, 5000 and 25000 items. Your program should populate those arrays with randomly generated integers with a value between 0 and the 2n where n is the size of the array. Create an implementation for the following sort operations.

* 1. Bubble sort
  2. Insertion sort
  3. Merge-sort
  4. Quicksort
  5. Heap-sort
  6. Counting sort
  7. Radix-sort

**Task 2:**

1. Test each of the sort operations and record the time the sort takes to complete. You should test each on the same unsorted array to get the best comparison. You should do this for each array size (from task 1) a minimum of 10 times.
2. If any 1 test run takes longer than 5 minutes, you may discontinue that test and record that the time took longer than 5 minutes. Likewise, should any test crash due to running out of memory, record that as well.
3. Explain how well or poorly it matches your expectations for performance. Include in the lab report a table of performance comparison among various sorting techniques.

**Task 3:**

1. Create a linked list class that can store Student information.
2. The Student information must include First Name, Last Name, MNumber and may include any other additional information you wish.
3. In addition to any required members to make the linked list work, include any 3 of the above sort algorithms to support sorting by First Name, Last Name and MNumber (each algorithm sorts by 1 of the 3) and includes a parameter for ascending or descending.
4. Seed your linked list with data for 20 students.
5. Include a menu interface that allows the user to select the sort method and direction and will display the sorted students on the screen.
6. Include in the lab report screenshots of input and outputs from various sorting techniques.

**Lab Submission:**

1. Write a lab report including the following information:
   1. The sections from each task indicated to be included in the lab report.
2. Include input and output files (if any), and any special instructions to compile and run those programs.
3. In a group project, submissions must include what each group member has contributed.

Package all files in a single zip folder and upload the file to canopy website within the specified due date.

**Lab Grading:**

1. 10% - Lab attendance
2. 35% - Task 1 has been correctly implemented and meets all requirements.
3. 20% - Task 2 has been correctly implemented and meets all requirements.
4. 15% - Task 3 has been correctly implemented and meets all requirements.
5. 20% - Lab report contains all required information and is well written.

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