Lab 4 – Sorting Objects

In class, we looked at three efficient sorting algorithms – the Merge sort, Quick sort, and Heap sort. The implementations demonstrated in class or included in the lecture slides were used to sort arrays of integers, but, of course, integers are not the only things that need sorting. When working with objects, the same sorting algorithms can often be applied. In this lab, you will implement two of the three algorithms we examined, and use them to sort an array of Objects.

EXAMPLE PROGRAM

While there is no example program for this lab, referring to the implementations of all three sorting algorithms as provided in the lab should give you some guidance.

YOUR PROGRAM

For your lab, you will need to define a class Student, with private members for first name, last name, and GPA, and any methods you determine that you need (constructors, gets/sets, etc.) Once the class is defined, you will need to populate an array of Student objects to be sorted with data provided in the students.txt file. The format of this data file is as follows:

- The first line contains an integer, which is a count of the number of student records contained in the file. You can use this value to define the size of the array.
  - Note: When I test your code, I may not use the same input file, so you will need to dynamically size the array based on the file input.
- Every line after the first contains three pieces of information, separated by spaces:
  - First Name, Last Name, and GPA

Once the array is populated with the student data, we are ready to start sorting! Choose two of the three sorting algorithms we discussed in class (MergeSort, QuickSort, and HeapSort), and implement them to sort the Student data by GPA.

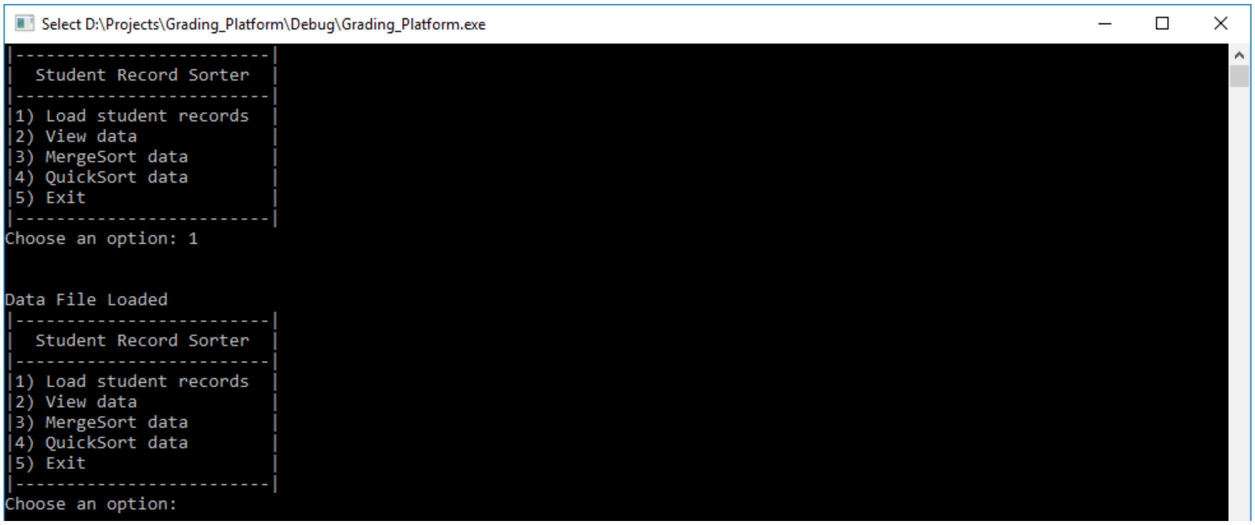
To test your code, and prepare it for submission, create a menu-based driver that provides 5 options:

1. Load the Data
   a. This will load the data from the text file. Once the data has been sorted, this option will also reload the data, so it can then be sorted again with the second method for testing.
2. View the Data
   a. This will just print out the current contents of the array, either sorted or unsorted.
3. Sort using the first algorithm
4. Sort using the second algorithm
5. Exit the program

An example of this can be seen in the sample screenshots below. When you are done, submit the completed .cpp file (or files, if using additional files for the class) through the Blackboard submission tool. Name the source file [FirstInitial][LastName]_Lab4.cpp. For instance, my submission would be LSilcox_Lab4.cpp.

SAMPLE SCREENSHOTS

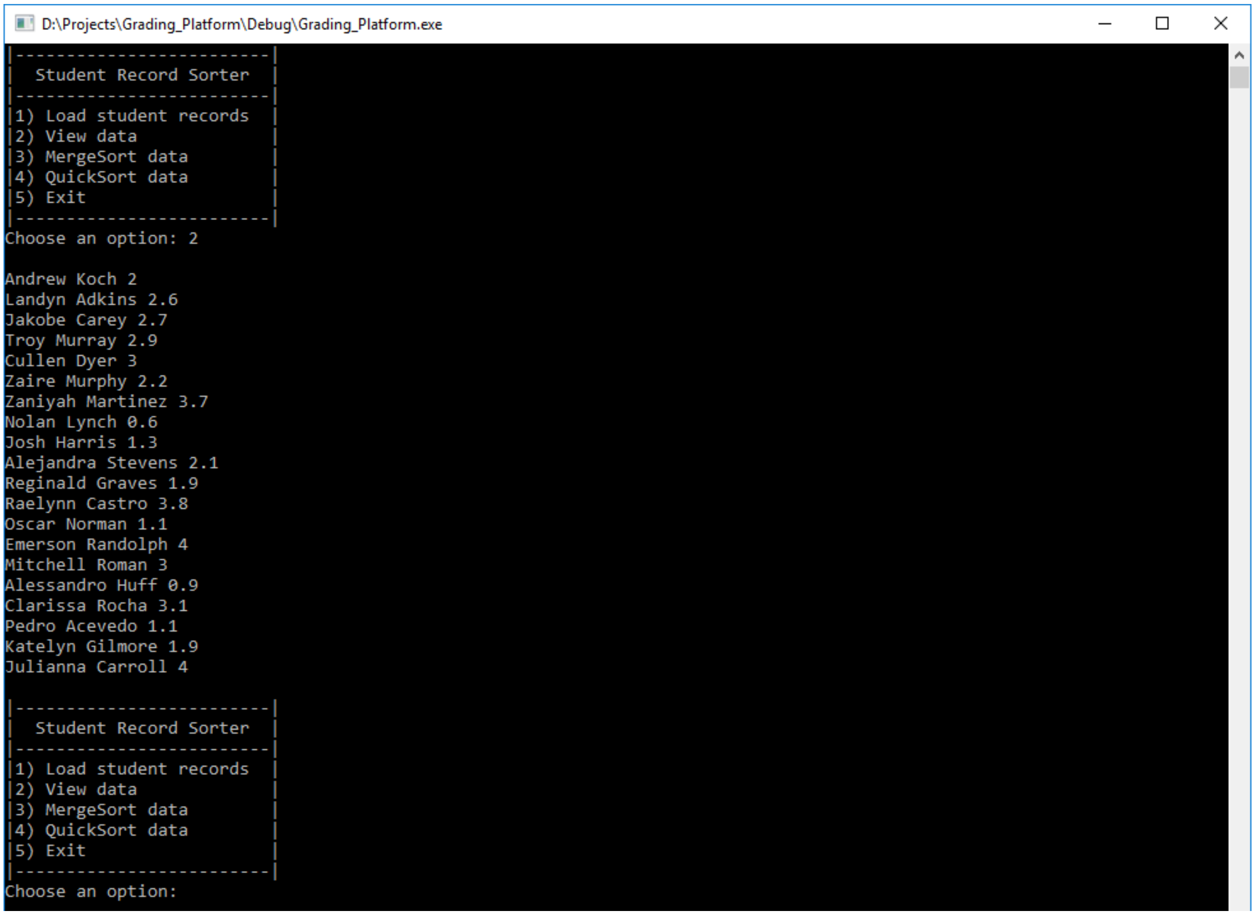
Loading the Data


Select D:\Projects\Grading_Platform\Debug\Grading_Platform.exe

```
|------------------------|
|  Student Record Sorter |
|------------------------|
|1) Load student records |
|2) View data            |
|3) MergeSort data       |
|4) QuickSort data       |
|5) Exit                 |
|------------------------|
Choose an option: 1


Data File Loaded
|------------------------|
|  Student Record Sorter |
|------------------------|
|1) Load student records |
|2) View data            |
|3) MergeSort data       |
|4) QuickSort data       |
|5) Exit                 |
|------------------------|
Choose an option:
```

Viewing the Data (originally unsorted)

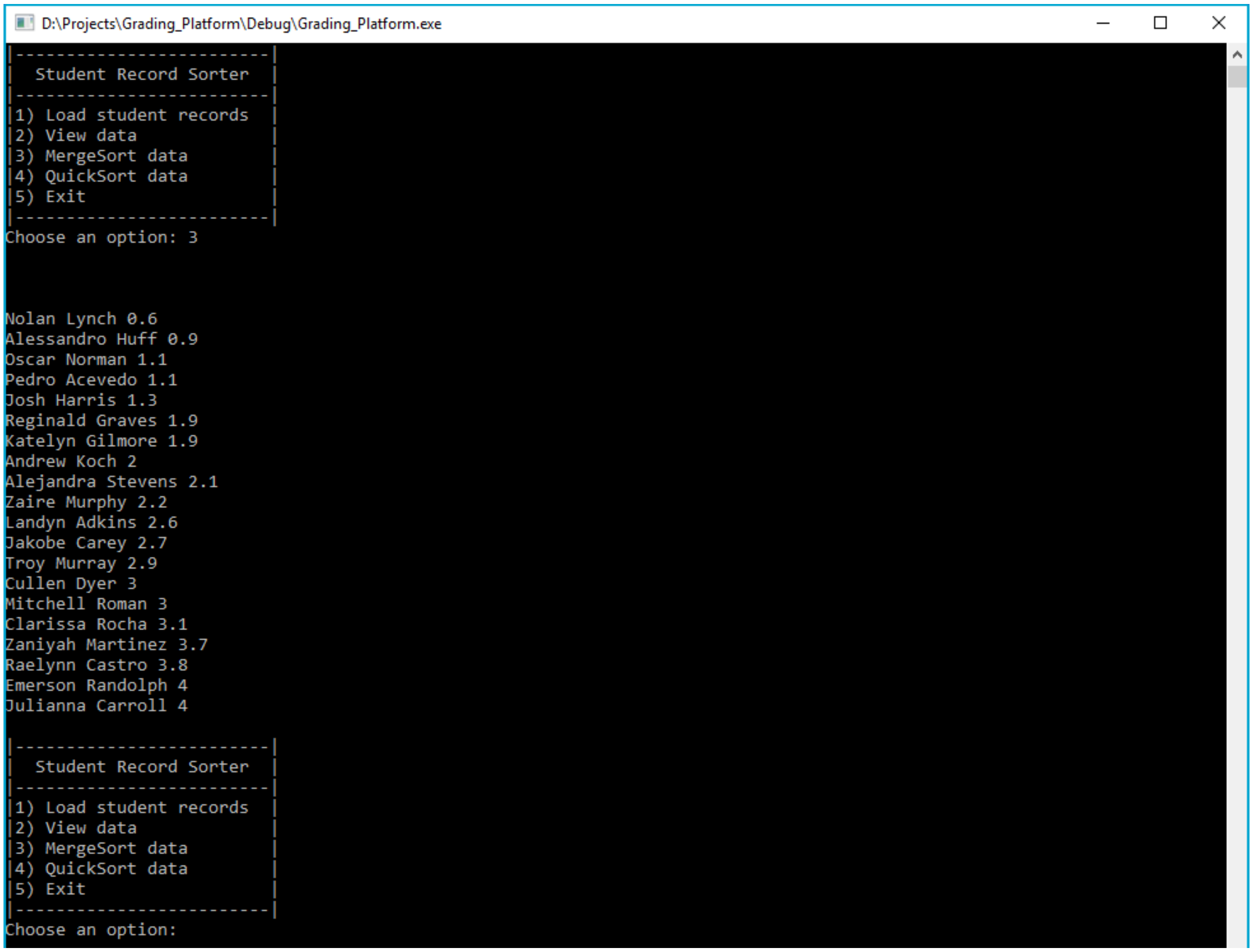


```
|------------------------|
|  Student Record Sorter |
|------------------------|
|1) Load student records |
|2) View data            |
|3) MergeSort data       |
|4) QuickSort data       |
|5) Exit                 |
|------------------------|
Choose an option: 2

Andrew Koch 2
Landyn Adkins 2.6
Jakobe Carey 2.7
Troy Murray 2.9
Cullen Dyer 3
Zaire Murphy 2.2
Zaniyah Martinez 3.7
Nolan Lynch 0.6
Josh Harris 1.3
Alejandra Stevens 2.1
Reginald Graves 1.9
Raelynn Castro 3.8
Oscar Norman 1.1
Emerson Randolph 4
Mitchell Roman 3
Alessandro Huff 0.9
Clarissa Rocha 3.1
Pedro Acevedo 1.1
Katelyn Gilmore 1.9
Julianna Carroll 4

|------------------------|
|  Student Record Sorter |
|------------------------|
|1) Load student records |
|2) View data            |
|3) MergeSort data       |
|4) QuickSort data       |
|5) Exit                 |
|------------------------|
Choose an option:
```

Sorting the Data



```
|------------------------|
|  Student Record Sorter |
|------------------------|
|1) Load student records |
|2) View data            |
|3) MergeSort data       |
|4) QuickSort data       |
|5) Exit                 |
|------------------------|
Choose an option: 3



Nolan Lynch 0.6
Alessandro Huff 0.9
Oscar Norman 1.1
Pedro Acevedo 1.1
Josh Harris 1.3
Reginald Graves 1.9
Katelyn Gilmore 1.9
Andrew Koch 2
Alejandra Stevens 2.1
Zaire Murphy 2.2
Landyn Adkins 2.6
Jakobe Carey 2.7
Troy Murray 2.9
Cullen Dyer 3
Mitchell Roman 3
Clarissa Rocha 3.1
Zaniyah Martinez 3.7
Raelynn Castro 3.8
Emerson Randolph 4
Julianna Carroll 4

|------------------------|
|  Student Record Sorter |
|------------------------|
|1) Load student records |
|2) View data            |
|3) MergeSort data       |
|4) QuickSort data       |
|5) Exit                 |
|------------------------|
Choose an option:
```