L A B 2 G R A D I N G S H E E T

Sidney Sanders

Student Name:

1003

CS2020-Section:

We will grade the following items for this lab exercise. Make sure to hand in this grade sheet complete with your name and section number at the end of the lab period. Without a grade sheet, your lab will NOT be graded.

\_\_\_/1 Grade sheet uploaded to Canvas

\_\_\_/1 Photo file contains all commands specified, required files

(**lab2.cpp, lab2.log**) are in your class account.

\_\_\_/2 Program documentation header, function documentation headers, white space, meaningful variable and constant names

\_\_\_/1 List displayed before and after sort.

\_\_\_/10 Functions to sort/search:

\_\_\_\_\_\_\_ (5) Bubble sort algorithm used to sort charge numbers, sorts

correctly, list is displayed before and after sort.

\_\_\_\_\_\_\_ (5) Binary search algorithm used to search for charge numbers,

searches correctly when item is found and when it is not.

\_\_\_/10 Answer the following questions:

|  |  |  |
| --- | --- | --- |
| Pts | Question | Answer |
| 2 | What are advantages & disadvantages of linear search? | Advantage: Simple  Disadvantage: Not efficient |
| 2 | What are advantages & disadvantages of binary search? | Advantage: Much faster than linear  Disadvantage: More complicated |
| 2 | What would you need to change in your code to sort your data in descending order? | Add a sorting algorithm and add a temp variable to hold a value when swapping values |
| 2 | Would your binary search work if data is in descending order? Explain what changes, if any, would be necessary. | Yes, you would have to make sure the comparisons are correctly placed so the variables compared are correct. |
| 2 | Assume an array of size 125,609 elements is to be searched using linear and binary search. Compute number of comparisons required for each algorithm in worst case, i.e., item is not found in the array. | In binary search, will be only about 6 searches using the logN+1 calculation.  In linear worst case would be 125,609 comparisons. |

\_\_\_/20 **Total points**