Name:

|  |  |  |
| --- | --- | --- |
| Problem | Points | Score |
| 1 | 50 |  |
| 2 | 50 |  |
| Total | 100 |  |

Notes:

1. The first step in this exam is to create a workspace in the following directory:

/data/courses/ece\_1111/current/exams/ex\_01

Your directory should be your last name all lowercase, followed by an underscore, following by your first name (e.g, “picone\_joseph”). Set the permissions using “chmod u+rwx,g-rwx,o-rwx <lastname>” so only you have read and write permission to this directory. Create subdirectories within this directory: p01, p02, … You will use these for problems 1 and 2, … respectively. Put ALL your code in these directories. Do not touch your files after the exam is over.

1. Your code must be nicely formatted and well commented, use Makefiles, etc. as we have done all semester. Failure to do these things will significantly lower your grade.
2. You can use code I wrote in class (hint!) or code you submitted for your homework and lab assignments (hint!!!!!).
3. For this exam you are allowed to open a terminal window on your computer, you are allowed to web surf with Google, but you cannot use online chat or other interactive services, such as ChatGPT. Trust me – if you use that tool, I’ll probably be able to figure it out ;)

**(50 pts) Problem No. 1**:

Create a file named data.txt with four lines of text:

A is for apple.

b is for banana.

C is for cantaloupe.

g is for grapes.

Assume the length of a line is exactly 99 bytes – no more, no less.

Write a program that does the following:

1. Creates an array that is a fixed dimension of N lines x 99 characters (e.g., char[N][99]).
2. Read the file line by line into this array.
3. Prints the contents of the array to stdout.
4. Sorts the array using the following sort criterion:
* Case insensitive
* Compares strings based on the number of alphabetic characters in the string
(e.g., characters in the range [A-Z] or [a-z]).
* Orders the sort in descending order – longest strings first.

Your program should work for any file, not just the example above.

**(50 pts) Problem No. 2:**

You work for the famous ACME Corporation as a software engineer. Before there was the Internet, there was ACME. If your goal was to capture the speed roadrunner, ACME was your go-to mail order company (<https://www.youtube.com/watch?v=9m7evoFF83c>... I know, a very dated reference ☺

Your job today is to write a program that reads a list of customers for ACME and stores it in a linked list. Your input file contains data of the form:

* Customer name
* Customer address
* <blank line>

Once you read the data into memory, you should close the input file and display the contents of the linked list in a human-readable form.

An example input file might look like this:

Wile E. Coyote

101 Desert Valley, Death Valley Junction, California, 92328

The RoadRunner

123 Beep Beep Drive, Hollywood, California, 33004

The output would look like this:

Customer Number: 1

Customer Name: Wile E. Coyote

Customer Address: 101 Desert Valley, Death Valley Junction, California, 92328

Customer Number: 2

Customer Name: The RoadRunner

Customer Address: 123 Beep Beep Drive, Hollywood, California, 33004

You must represent each customer record as a structure, and you must store the customer records in a linked list.