Programming Assignment #2

Research Lab Access

CS 2308.003 and 004 Fall 2018
Instructor: Jill Seaman

Due: Wednesday, 9/26/2018: upload electronic copy by 9:00am!

Problem:

Write a C++ program that will allow a research lab manager to monitor the access to a research lab. Each member of the lab has access provided by their personal ID Card. Your program will provide information about the timing of the arrivals and departures of the lab staff.

Your program will be provided with a dataset where each row records an event, which is represented by the following information:
1) ID number - a positive integer number
2) Person's name - a string
3) Time of the event - a string in "01:23AM" format ("dd:dd:AM" or "dd:dd:PM")

Each event records the use of an ID card to either enter or exit the lab. A given data set corresponds to one day and is already sorted by ID number. You should assume that the lab is empty at the beginning of the day and events are consistent - a given person cannot enter or exit without the event being recorded, a given person cannot enter the lab twice without exiting once in between, etc.

The program should first read the events from a text file named "dataset.txt". It will contain up to 100 events. See the sample file on the class website.

Then, it should offer the user a menu with the following options:

1. Display the events sorted by ID number.
2. Display the events sorted by the time of event.
3. Print the person's name given his/her ID number.
4. Find out whether some person is still in the lab given his/her ID number.
5. Quit

The program should perform the selected operation and then re-display the menu. **Do not change the menu numbers** associated with the operations. Display an error message if the user enters an inappropriate number.

For options 1 and 2, display the information for each event on a single, separate line. The values should line up in columns (use setw). Headers for the table are optional.

For the **Lookup** operations, label the output values appropriately. For options 3 and 4, If the person is not found, display an appropriate message.

Additional Requirements:

- This program must be done in a **Linux or Unix** environment, using a command line compiler like g++. Do not use codeblocks, eclipse, or Xcode to compile.
- Your program **must compile** and run, otherwise you will receive a score of 0.
- The program must be **modular** (use top-down design), with significant work done by **functions**. Each function should perform a single, well-defined task.
- Use a **partially filled array** of structures to store the events**:**
  Use a counter variable to count the number of events that are read in from the file, and use this value as the size of the array for the search and sort functions.
- Your program should work for an input file with any number of events up to 100.
- You MUST use **binary search** for option 3, Lookup name by ID.
- You may use (and modify) the code from the book. See the Resources tool in TRACS. Please look at this code before you start implementing your program.
- I will put a sample input file on the class website (dataset.txt) and the console output from running my solution on that file (output2.txt).

Logistics:

Name your file **assign2_xxxxx.cpp** where xxxxx is your TX State NetID (your txstate.edu email id).

There are two steps to the turn-in process:

1. Submit an **electronic copy** using the Assignments tool on the TRACS website for this class (tracs.txstate.edu).
2. Submit a **printout** of the source file at the beginning of class, the day after the assignment is due. Please **print your name on top of the front page**, and staple if there is more than one page.

See the assignment policy on the course website (cs.txstate.edu/~js236/cs2308) for more details, including deadlines, penalties, and where to submit printouts outside of class.