CS 162 Computer Science II
Programming Assignment Project 2

Problem

You have noticed that there is a lot to do this term as we get familiar with C++ and programming on a larger scale. With a full load of classes, you may have many due dates to meet. Keeping track of everything is important. You have decided to write a program that reads tasks from the user and saves them in a data file to keep track of.

For version 1.1 of TaskTrack, we are going to read Task items from an external file when user starts the application and write all the Task items back to that file when user quits. You need to determine the file format when you write out the Task items so that you know how to read them back in next time when user starts the application.

Remember, your file must be a **text** file.

Requirements

Incorporate the use of C++ structs in your design.

You are *required* to use struct (no class or linked lists) to model the list and the item. Write an interactive text based menu interface (using a loop) that will allow the user to

- Enter a task or assignment
- Display all of the tasks that are in the file
- Find a task by Course
- Quit

For each task, you need to keep track of:

- Course Name that it is for (e.g., CS162)
- Description of the assignment (e.g., Finish Lab 2)
- Due date (e.g., 9/26/2009)

Allow the program to keep looping until user wants to quit. When the program starts, it should load the tasks from external file ("tasks.txt") into memory. When user enters the three items of a task, the program needs to read them in, save them in memory and eventually write them to the external data file ("tasks.txt"). The file format should look like: (The ';' is used as a delimiter or field separator.)

CS162;Finish Lab 2;9/26/2009
CS201;Take Quiz 1;9/28/2009

Design Requirements:

Please follow the below specifications and do not deviate from them. Failure to do this will result in deduction of points.

1. You must have a menu driven user interface with 4 items in the menu as mentioned above.
2. Write at least **four** functions WITH arguments for this assignment.
3. Use struct named Task to model task
4. Use array of structs to model the collection of tasks.
5. Hint: In this assignment, the description and course name may have multiple words in it. Therefore, you now SHOULD read using the 3 argument version of get.
6. Watch out. When using the 3 argument version of get you need to make sure to remove the delimiter or newline. Therefore, anytime you read (even a confirmation message), make sure to eat the newline!
7. Make sure to have a delimiter written between each item in the file – like a newline. This will be important when you read the information back from the file.

Do-Not List for All Projects in CS162:

- No Global Variables (you can have global constants)
- No use of the stdio library (use iostream and fstream)
- Instead of the string class, you will be using arrays of characters and the cstring library

Things You Should Do:

- Follow the style guide for this class
- Your programs should always guard against bad data being entered by mistake
- **Failure to follow the specifications will result in loss of points or 0 for your projects.**

Goals for This Project:

- Breaking tasks down into functions
- More about using character arrays
- File input/output

How to Submit Your Work:

The same as project 1 except you call it project2.tar