

CS 162 Computer Science II

Programming Assignment Project 3

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 1.2 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++ classes in your design.

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 could look like: (The ';' is used as a delimiter or field separator.) Please read the Design Considerations section on next page.

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

Design Considerations (Please read carefully and thoroughly):

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

1. Name your file project3.cpp. If there are multiple files, name them appropriately and name your main file project3.cpp.
2. Please be sure the source file includes your name, assignment description and number, and date, as a program comment!
3. Write at least **four** functions WITH arguments for this assignment.
4. **Use structs or class named Task to model task. You are encouraged in every way to use class to model task in this lab.** If you do the work here, you will benefit from this in projects 4 and 5.
5. **You MUST use class named TaskList to model the collection of tasks.**
6. **You MUST use array of Task to implement the above class, TaskList.**
7. When using class, please make sure you encapsulate the data, which means make all the instance data members private and provide accessor methods and mutator methods to access and manipulate the data.
8. 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.
9. 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!
10. 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.
11. Your txt file must have at least 3 line items when you submit it.

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
- No goto statements.

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 stick to above specifications will result in loss of points or a 0 for your lab.**

Goals for This Project:

- Using class to model Abstract Data Type
- OOP-Data Encapsulation

- 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 project3.tar