

CS250 Assignment 2

Structs

Date assigned: Wednesday, February 17, 2016

Date due: Monday, February 29, 2016

Points: 30

Goals for Assignment 2

1. Implement a C++ program that uses files, arrays, enum, and structs.
2. Implement error handling for you project
3. Build an ADT
4. Break up a program into well-defined functions. It is important that your functions be small and focused on a particular task.

You've been hired by the University to work on part of the course enrollment system. Your task will be to read course enrollment data from a file, print this information on request, and add students to courses.

You have an input file with the following information about the courses

- Prefix (only CS or MATH will appear in the file)
- Course Number
- Max Capacity
- Current enrollment

Now, you need to provide a menu of functions for users to interact with your software:

1. Print all courses
2. Print a single course
3. Add a student to a course
4. Quit

The program should continue to allow the user to enter menu choices and receive information until the user chooses to quit.

Input: The file will contain lines as follows. The last two integers on a line are capacity and current enrollment, respectively. You must be able to read 10 courses from the file. For example:

```
CS 250 48 36
CS 460 24 9
CS 494 24 9
MATH 121 25 23
MATH 240 30 20
```

Your **output** is to look exactly like the following:

```
*****  
PACIFIC UNIVERSITY COURSES  
*****
```

1. Print all courses
2. Print one course
3. Add a student to a course
4. Quit

Option: 1

```
CS 250 Cap: 48 Available: 12  
CS 460 Cap: 24 Available: 15  
CS 494 Cap: 24 Available: 15  
MATH 121 Cap: 25 Available: 2  
MATH 240 Cap: 30 Available: 10
```

1. Print all courses
2. Print one course
3. Add a student to a course
4. Quit

Option: 2

```
Prefix: MATH  
Number: 121
```

```
MATH 121 Cap: 25 Available: 2
```

1. Print all courses
2. Print one course
3. Add a student to a course
4. Quit

Option: 3

```
Prefix: MATH  
Number: 121
```

```
MATH 121 Cap: 25 Available: 1
```

```
. . . < add more students to MATH 121>
```

1. Print all courses
2. Print one course
3. Add a student to a course
4. Quit

Option: 3

```
Prefix: MATH  
Number: 121
```

```
Course already full.
```

1. Print all courses
2. Print one course
3. Add a student to a course
4. Quit

Option: 4

Notes:

1. Your main function is to be mainly variable declarations and function calls.
2. Test your program one function at a time.

To complete this assignment you must submit the following:

1. An electronic copy of your program on Grace

- a) Add your project to your existing Visual Studio Solution. Your project for this assignment must be named **02_Structs**. It is vital that you name your solution and your project correctly!
- b) Type your program (fully documented/commented) into the project. You need to follow the coding standards from the CS250 Web page. These coding standards have been modified to include additional C++ language features introduced in CS250, so please be sure to read the new coding standards.
- c) Pay attention to the example output. Your program's output must look **exactly** like the sample output. The spacing and newlines in your output must match exactly.
- d) Make sure that your program builds without errors & warnings and runs correctly. If you get any errors or warnings, double check that you typed everything correctly. Be aware that C++ is case-sensitive. You will lose 10% if there are any warnings and 70% if your program does not build successfully.
- e) Once you are sure that the program works, it is time to submit your program. You do this by logging on to Grace and placing your complete solution folder in the **CS250-0X Drop** folder.
- f) The solution must be in the drop folder by the time class starts on the day the assignment is due. Anything submitted after that will be considered late.
- g) **Error Handling:**
 - If the user provides an invalid selection for the menu, redisplay the menu with no error message
 - If the file contains a prefix other than MATH or CS, print an error message and terminate the program.
 - If the file contains more than 10 courses, print an error message and terminate the program.
 - If the enrollment is larger than the capacity, print an error message and terminate the program.
 - If the class is full, no student can be added.

2. A hard copy of your program

- a) The hard copy must be placed on the instructor's desk by the time class starts on the day that it is due.
- b) The hard copy must be printed in color, double-sided, and stapled in the upper left corner if your solution contains multiple pages.
- c) Your tab size must be set to 2 and you must not go past column 80 in your output.

Remember, if you have any problems, come to me straight away

with your project on Grace. Good Luck!!!! 😊