

PROGRAMMING ASSIGNMENT 4

CS1410 - 100 points

OUTCOMES

After you finish this assignment, you will be able to do the following:

- Define a class
- Define and overload constructors
- Define a destructor
- Define accessors and mutators
- Define member functions
- Instantiate objects
- Call member functions

DESCRIPTION

Write a class named **Course** with the following **private** data members.

- **crn**: The course reference number.
- **name**: The course name.
- **creditHours**: The course number of credit hours.
- **year**: The course year.
- **semester**: The course semester (Spring, Fall, or Summer).

The class should have the following **public** members:

- **constructor #1**: The default constructor which sets all numeric data members to 0's and all string data members to empty strings. Use an initializer list for this.
- **constructor #2**: Accepts arguments for the course crn and name and sets the remaining numeric data members to 0's and the remaining string data members to empty strings. Use an initializer list for this.
- **constructor #3**: Accepts arguments for the course crn, name, credit hours, year, and semester. Use an initializer list for this.
- **destructor**: An empty destructor.
- **accessors and mutators**: Write a getter function (an accessor: **getCrn**, **getName**, ...) and setter function (a mutator: **setCrn**, **setName**, ...) for each data member. Note that in C++:
 - Getter functions should be **const** functions with no arguments; they also should return **const** references if their data members have non-primitive data types.
 - Setter functions accept single arguments and if the data member has a non-primitive data type, the argument should be a **const** reference.
- **str**: A **const** function that returns a string representation of the course including its crn, name, year, semester, and credit hours. This function should use **stringstream** to piece together the returned string. The format of the returned string is left to you.

In the **main()** function:

- Instantiate three objects of this class using the **new** operator: one object for each constructor. On these objects, call the mutator functions to supply values (of your choosing) for the data members that are initialized to 0's or empty strings by the constructors.
- Call the **str** function on each object and print the returned string to the console.
- Finally, delete these objects and return them back to the operating system.

HINT: Check out worksheets 7 and 5 to learn about classes and objects and **stringstream**.

INSTRUCTIONS

For this assignment, you need to have a GitHub account. If you don't have one already, please sign up for one at <https://github.com/>.

Getting the assignment starter code from GitHub:

- Sign in to GitHub.
- Go to the assignment link <https://classroom.github.com/a/j2MgvlOq> and accept the assignment. This should create a private repository under your GitHub username for this assignment. Click on the given link to open this repository and see the starter code.
- Click on the **Clone or Download** button dropdown and copy the given URL.
- Navigate to your assignments folder (or any folder you want this assignment to be placed in) and open it using Visual Studio code.
- In Visual Studio Code, open a new terminal and then run:

```
wsl (for Windows 10 only)
```

```
git clone THE_URL_YOU_COPIED
```

This will download the starter code of this assignment from GitHub and create a folder for it with a name like **cs1410-assignment-04-github_username**. This is the folder where your program file(s) (**.cpp** and/or **.h**) should reside.

- Open the assignment folder (whose name looks like **cs1410-assignment-04-github_username**) in Visual Studio Code and start writing your program.

Compiling your C++ program:

- From inside the assignment folder in Visual Studio Code, open a new terminal and run:

```
wsl (for Windows 10 only)
```

- To compile your program run:

```
make
```

This command will call the C++ compiler on your program, compile it, and, if no compilation errors are found, create an executable program named "**run**" for it. If there are compilation errors, read the console error messages and then go back to your source files (**.cpp** and/or **.h**) and fix them. Save your changes and run the "**make**" command to compile the program again.

- To run your program, run:
`./run`
- To clean (remove) old compilation files and start over, run the command:
`make clean`
You can now run the "`make`" command to compile your program again and the "`./run`" command to run it.

Submitting your program to GitHub:

- Make sure to save your changes and commit them to GitHub when you are done. You can do that by running the following commands from inside your assignment folder:
`wsl` (for Windows 10 only)
`git add .`
`git commit -m 'short commit message goes here'`
`git push`
Make sure to do this at least once by the deadline. For your final submission, I recommend using "**Final submission**" for the commit message. **Note that committing changes is not enough; you have to push them to GitHub; otherwise, your changes will stay on your local machine and I will not be able to see your submission.**
- Go to your assignment repository in github.com and make sure your changes are there.
- Click on the **Clone or Download** button dropdown and copy the given URL. Go to Canvas and submit the copied URL. **This URL must be submitted in Canvas after you make your "Final submission" to GitHub.**

RUBRIC

CRITERIA	POINTS
Class definition	10
Constructors	15
Accessors and mutators	30
Member functions	10
Destructor	5
Object creation and member function calls	15
Readable, commented, and properly indented code	10
TOTAL	100