Due: 7/6 (11:59PM)

You are to code the following three source files in C++:

- **Student.h**
  - In this file, you declare a class named *Student*.
  - This class contains 4 **private** attributes, *name*, *idNumber*, *department*, and *year*.
  - *name* is a `struct` that contains two strings: *firstName* and *lastName*.
  - *idNumber* is an integer and *department* is a string.
  - *year* is an `enum` that contains four possible values: *FRESHMAN*, *SOPHOMORE*, *JUNIOR*, *SENIOR*.
  - Note that `struct` and `enum` declarations should be placed before the class declaration.
  - Define public getter functions and setter functions for all 4 attributes, that is, 4 getters and 4 setters.
- **Student.cpp**
  - In this file, you provide definitions for 3 constructors of *Student*.
  - The first constructor takes 4 parameters, one for each attribute, then initializes the attributes accordingly.
  - The second constructor takes 2 parameters, one for *name* and one for *idNumber*, then initializes those attributes accordingly. The remaining 2 attributes are initialized as *“”* (empty string) and *FRESHMAN*, respectively.
  - The third constructor is a default constructor that takes no parameters. Here, the 4 attributes are initialized as *“”* (*firstName*), *“”* (*lastName*), 0 (*idNumber*), *“”* (*department*) and *FRESHMAN* (*year*), respectively.
- **hw3.cpp**
  - In this file, you define main function.
  - In the main function, first create at least 3 *Student* objects.
  - The first object must be created using the first constructor (4 parameters).
  - The second object must be created using the second constructor (2 parameters).
  - Since the second constructor does not provide proper information for *department* and *year*, set those values by calling their associated setter functions.

- The third object must be created using the default constructor.
- Since the default constructor does not provide proper information for any attributes, set all 4 values by calling their associated 4 setter functions.
- After creating all the objects, call a function named *displayStudent* for each student to print all the information (all 4 attributes) about the student. Note that this function must be called at least 3 times to print all objects. You must define this *displayStudent* function (next to *main*) with the following prototype:

  ```
  void displayStudent(Student);
  ```

- When displaying *year*, make sure the freshman year starts at 1, not 0.

Example run:

```
Name: Roger Federer
ID Number: 12345
Department: Art
Year: 4

Name: Rafael Nadal
ID Number: 56789
Department: Computer Science
Year: 3

Name: Novak Djokovic
ID Number: 13579
Department: Physics
Year: 1
```

What to submit:

- All source files (.cpp, .h) needed for compilation

How to submit:

- Use Canvas Assignment Submission menu to submit the assignment electronically at Canvas.
- Make sure to zip all your files into `hw3.zip`, then submit your `hw3.zip` as a single file.

Policy

- Make sure all your C++ programs properly compile and run on Eclipse C++.
- Projects will be graded 20% on style/standards and 80% on proper execution. Make sure to follow the *Coding Standards* posted on the course webpage. In particular, use proper indentation on each line of your code.
- At the beginning of each file (.cpp, .h), provide comments specifying the author, date, and a brief description of the file.
- Each source file (.cpp, .h) must contain enough comments here and there to make it easy to follow your code. Insufficient comments could lead to loss of points.
- Non-compilable program will get almost no credit (e.g., executable code not produced due to compile errors).
- Non-working program will get almost no credit (e.g., the executable is terminated immaturely due to run-time errors).
- Copying other's code is strictly prohibited. If identical (or nearly identical) submissions are found among students, every student involved will get automatic zero for the assignment. The same goes for copying existing source code from the internet.