

CMPE 50, Object-Oriented Concepts and Methodology, Spring 2015, Tarnag

Lab #5 – Introduction to Classes

Exercise 1: Write the definition of a class representing a student record:

The student record class should have member variables for all the grading components, namely:

- 2 quiz scores, graded on the basis of 10 points, weighted at 25%
- 1 midterm exam, graded on the basis of 100 points, weighted at 25%
- 1 final exam, graded on the basis of 100 points, weighted at 50%
- the student's weighted average numeric score

Make all member variables private.

(note that at this point, you don't have to worry about assigning values to the member variables – you are just declaring them, creating placeholders for future objects of this class; you need, however, to make appropriate decisions regarding the data type of each variable)

- Include member functions for each of the following:
 - a) Member functions to *set* each of the member variables to values given as arguments to the function, i.e., the *mutators*
 - b) Member functions to *retrieve* the data from each of the member variables, i.e., the *accessors*
 - c) A default constructor that gives default values to the objects of the class when initializing them.
 - d) A void function, i.e., a function that has no return value, that calculates the student's weighted average numeric score for the entire course and sets the corresponding member variable

Test your code using a simple test program.
(based on Progr. Proj. 10.2)

Exercise 2:

Define a class for a type called CounterType. An object of this type is used to count things, so it records a count that is a nonnegative whole number. Include the member variable(s) you think are appropriate for this class. Also, include the following member functions:

- a) a default constructor that sets the counter to zero
- b) a constructor with one argument that sets the counter to the value specified by its argument
- c) a function used to increase the count by one
- d) a function used to decrease the count by one
- e) a function that returns (get function) the current count value
- f) a function for that outputs the count to a stream. This output function should have one formal parameter of type ostream for the output stream that receives the output (note that this can be cout or an output file).

Be sure that no member functions allow the value of the counter to become negative (Provide appropriate messaging in case the value becomes zero).

Embed your class definition in a test program.