**Take-Home: Quiz 3 (15 pts) – More OOP**

Using Canvas <https://canvas.wsu.edu/>, please submit your solution to the correct quiz folder. Your solution should be a .pdf file with the name <your last name>\_quiz3.pdf and uploaded. To upload your solution, please navigate to your correct Canvas ***lab*** course space. Select the “Assignments” link in the main left menu bar. Navigate to the correct quiz submission folder. Click the “Start Assignment” button. Click the “Upload File” button. Choose the appropriate .pdf file with your solution. Finally, click the “Submit Assignment” button.

1. **(4 pts)** What is a *constructor*? Explain.

**A constructor is a special member function in a class, which is named the same as the class name. This function initializes the data members of the object and does not return any value. It is called implicitly when an object is instantiated. If a class does not explicitly provide a constructor, then the compiler provides a default constructor with no parameters.**

1. **(4 pts)** What is *encapsulation*? Explain.

Encapsulation is a way of organizing or wrapping of data/attributes and methods/operations into a structure. It is demonstrated by object (an instantiation of a class). Good encapsulation is to prevent the access to the data directly, the access to them is provided through the functions of the class. For example, there are two different departments like the Computer Science and the Physics. Each department keeps only records and related information of itself. So in order to access Physics records, the Computer Science must contact and request the Physics to give access to the particular record.

1. **(4 pts)** What is a *reference*? Explain.

**A reference is another name for an existing variable. If a reference is set to a variable, then it can access and modify the variable. When we pass parameters by reference, no copy of the contents is made and the called function can access the data directly, and modify them.**

1. **(3 pts)** What is a class? Explain.

A class is an object-oriented concept which encapsulates data and procedural abstractions to represent a real-world entity. A class consists of data members (attributes) and member functions (operations). For more security, we should make the data members private (information hiding), and then they may only be accessed by member functions of the class (or friends). This is a well-designed class.