CptS 122 – Data Structures
February 16, 2024

Your Name: ___________________________ TA's Name: ___________________________
ID#: ___________________________ Section #: ___________________________

SOLUTION: Take-Home: Quiz 4 (15 pts) - OOP and C++

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>_quiz4.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 *class*? Explain.

   A *class* is an object-oriented (OO) concept which groups (encapsulates) data and procedural abstractions to represent a real-world entity. Data abstractions that are hidden (information hiding) in the class are surrounded by procedural abstractions. In a well-designed class, the only way to access the data/attributes is to use the well-defined procedures/functions/methods that are available.

2. **(4 pts)** What is a *copy* constructor? Explain.

   A *copy constructor* makes a *copy* of an object of the same type. A copy constructor is *implicitly* invoked when an object is *passed-by-value*! A *shallow* copy is made if only the data members are copied directly over to the object. A *deep* copy is made if new memory is allocated for each of the data members. For example, if we have a linked list of nodes that were allocated on the heap, to make a *deep* copy we would need to allocate new nodes and copy the data from the first list's nodes into the newly allocated nodes belonging to the second list. A *shallow* copy would just copy the head pointer member of the first list into the head pointer member of the second list.

3. **(4 pts)** What is *encapsulation*? Explain.

   ***Encapsulation* is a way of organizing or wrapping of data/attributes and methods/operations into a structure (or capsule). Objects naturally impose encapsulation - attributes and operations are closely tied together.**

4. **(3 pts)** What is *function overloading*? Explain.

   ***Function overloading* is the capability to define more than one function with the same name as long as these functions have a different set of parameters, including number and/or types.**

Instructor: Andrew S. O'Fallon