CptS 122 – Data Structures

October 9, 2017

Your Name: Garrick Hutcheson-Geschickter TA’s Name:Collin Westenhaver

ID#: 11572317

**Take-Home: Quiz 4 (15 pts) – More OOP**

**Print out, and provide your solutions to your TA in lab this week!**

1. **(4 pts)** What is *information* *hiding*? Explain.

Information hiding is the notion of giving out information about the data or algorithms in a program on a “need to know” basis. If a programmer using the code does not need to know about all the intermediate values that are used to make a function work then don’t let them see or modify them. Information hiding is essential in discerning the way in which a segment of code was *meant* to be used. By making some things hidden it stops people from using the code in a different way.

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

Streams are the means by which programs read and write data to and from different sources. ~~Many choose to describe a stream using the analogy of a conveyor belt but the image of a stream was chosen as an analogy in and of itself. One may stand by the stream and take “water” or data out as it goes by. You may store it in different “buckets” or data types or choose not use it at all.~~ Honestly I’m not even gonna try to make a good analogy. Streams are used to send and receive data in a sequential order from different specified sources. As the programmer it is up to us to make sure any data entering the stream is properly formatted for use on the other side, whether that other side is a text file in local memory or string on the stack of the program. In C++ one may overload the extraction and insertion operators, which pull data from and add data to streams, to do this formatting for us.

1. **(4 pts)** What is *procedural* abstraction? Explain.

Procedural abstraction, is the notion of combining all of the subtasks needed to complete a larger task under one name. Much like information hiding, procedural abstraction hides the details of implementation from people using the algorithm. Procedural abstraction allows for us to perform a task that requires several operations in a single function call.

1. **(3 pts)** What is an *object*? Explain.

An object is a data structure that ties data, and the functions used to manipulate or use said data, together. Objects are instantiations of a class. This means that there may be any number of objects of the same type and that we may define member functions that operate directly on an object. We may also define functions that accept objects as parameters, which is a strong tool as it allows to pass data and the functions used to manipulate that data in to a function simultaneously.