**Take-Home: Quiz 9 (15 pts) – More OOP!**

**Please note: this quiz is due next week by Wednesday, December 8. You don’t have to submit it this week, if you don’t have time to finish it!**

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>\_quiz9.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. **(5 pts)** What is an abstract class (2 pts)? What is the purpose of an abstract class (3 pts)?

An abstract class is designed with at least one pure virtual function. We cannot declare an object of an abstract class, as well as we cannot use it as a parameter type, or a function return type. However, we can define pointers or references to an abstract class.

Example:

//Abstract class

class Animal

{

virtual void cries() = 0;

}

The purpose of an abstract class is to provide a common base class for a set of concrete subclasses. Through inheritance from that abstract base class, derived classes are formed that operate similarly.

For example, we are going to implement a zoo of 10 different types of animal. It is not good to just define 10 arrays of those animal. Instead, we can just define only one array of pointers to the abstract class Animal. Through each pointer, we just instantiate one type of animal as we need. Then calls cries() function for each element in the array. Each will respond to the function in a unique way since each derived class provided its own implementation for it.

1. **(5 pts)** What is Big-O notation? Discuss space and time complexity.

Big O notation is a mathematical notation that measures the complexity an algorithm may acquire, which is relative to its input size of one or more parameters.

Space complexity is the total space taken by the algorithm with respect to the input size.

For example:

The constant complexity (O(1)): The algorithm takes the same amount of space regardless of the input size.

The linear complexity (O(n)): The algorithm takes space directly proportional to the input size.

Time complexity is to measure the amount of time taken by the algorithm according to the input size. It may evaluate how efficient the algorithm is.

For example:

Constant time – O (1): The runtime will always be the same, not dependent on the input size n.

Quadratic time – O (n2): The runtime grows proportionally to the square of the input size.

1. **(5 pts)** What is a C++ exception? Explain.

**An exception is a response to an exceptional circumstance that arises while a program is running. It provides a mechanism for building robust and fault-tolerant programs, since it throws an exception when a problem shows up, which prevents the program from just stopping execution without chance to recover.**

**For example,**

**We declare a vector of two elements. If the element at index 2 is accessed, it will raise an error, and stop executing the remaining code. So we will throw an exception there.**

**vector<int> arr = { 0, 5 };**

**try { cout << arr[2] << endl; }**

**catch (exception& ex) { cout << “Exception occurred!” << endl; }**