**Computer Science II – Weekly Quiz 03/24/2017**

**Each question below is worth 10 points.  Please answer clearly in the space provided.**

**Jake Lorah**

1. What does the acronym API stand for?  Explain the main goal of API in computer programming and describe how to access it in Greenfoot.

API stands for Application Programmers Interface. The main goal of it is to allow the program to access another software application of a Web based service or tool. Also it makes the process easier for other programmers. In Greenfoot, it is called Greenfoot Class Documentation, and you can access it under the help tab.

1. List four things that Greenfoot’s version of API contains.

The four things that Greenfoot’s version of API contains is all available classes, all available methods per given class, descriptions of each, and corresponding return types.

1. Describe the purpose of implementing random behaviors in Greenfoot animations.

The purpose of implementing random behaviors in Greenfoot animations is to make the objects in the animation behave more life-like.

1. Please review the 5 method names below.  Explain whether each one is named according to method naming rules or not:
   1. makeTheWobatTurnRigt();

This method name is following the rules because there are no spaces or special characters. Also the first word is lowercase, and every word after that is upper case.

* 1. atWombatWorld’sEdge();

This method name is not following the rules because there is a special character in it, and those are not allowed for method names. The apostrophe in Worlds can’t be there.

* 1. GenerateRandomNumbers();

This method name is not following the rules because the first word is capitalized. The first word must always be lowercase, but every word after that is capitalized.

* 1. make The Wombat Turn Right();

This method name is not following the rules because there are spaces in it. There must be NO spaces in your method name.

* 1. getTotalNumberOfLeavesEaten();

This method name is following the rules because there are no spaces or special characters. Also the first word is lowercase, and every word after that is capitalized.

1. Describe the purpose of Java’s Equality and Relational Operators and provide 3 examples.

Java’s Equality and Relational Operators determine how the values in question relate to each other. 3 examples are…

== equal to

!= not equal to

> greater than

1. What is the main benefit of Java over Javascript as far as creating new methods is concerned?

In JavaScript, you can’t edit/create your own method. In Java, you are allowed to create your own method and you can edit it to make it do what you like.

1. Explain the complete process of generating random behaviors in Greenfoot.  Compare with the same process in Javascript.

Greenfoot allows you to create random behaviors by using random numbers generated through one of the methods associated with the Greenfoot class. This can be accomplished using the getRandomNumber method that is built into the Greenfoot environment. When you provide a parameter value for the upper limit, this method will return a number between 0 and whatever the limit is. In JavaScript, to generate a random number, you use Math.Random.

1. Describe the significance of the keyword static in Java programming.  Where would you find this keyword?

Static methods are a collection of methods found in the same class. All methods belonging to a particular class are marked with the keyword static at the beginning of the signature.

1. What is dot notation?

Dot notation is the process of calling objects from other classes by the class name and method name.

1. Explain when Java requires the use of dot notation and when it is optional.

Dot notation is required if the method we wish to call is in a different class or hasn't been inherited from the sub-class. Then the name of the class or the object containing the method must be specified, which you then must use dot notation for. Dot notation is optional when you just want to organize your code a bit more, and easily call methods in other Java classes and objects.