

## COMP 1409 Introduction to Software Development 1 Quiz #8 Suggested Solution

1. Below is the beginning of a class called School which stores Student objects in an ArrayList. You fill in the missing bits according to the comments provided. Do not remove the point designations for each section.

```
import java.util.ArrayList; // write the import statement here ( 1 point )
public class School
{
    // declare a variable that is an ArrayList of Student ( 1 point )
    private ArrayList<Student> allStudents;

    public School()
    {
        // initialize the instance variable to hold a new ArrayList of Student ( 2 points )
        allStudents = new ArrayList<Student>();
    }

    public void enrollStudent(Student theStudent)
    {
        // add a student to the the array list ( 2 points )
        allStudents.add(theStudent);
    }
}
```

2. The Student class has these fields with appropriate accessor methods: firstName, lastName, studentNumber. **You are not required to write this class.** Write a method for the `School` class that displays on the screen the first and last name of each student in the school. This method must use a for-each loop. (4 points).

```
public void displayStudents()
{
    for(Student aStudent : allStudents) {
        System.out.println(aStudent.getFirstName() + " " + aStudent.getLastName());
    }
}
```

3. Assume the ArrayList is called **allStudents**. What is wrong with the following method? Explain. ( 2 points )

```
/** Display the last name of the student at the index position
specified by the parameter */
public void displayStudentName(int index)
{
    int numberOfStudents = allStudents.size();
    if((index >= 0) || (index < numberOfStudents)) {
        Student aStudent = allStudents.get(index);
        System.out.println(aStudent.getLastName());
    }
    else {
        System.out.println("Not a valid index position.");
    }
}
```

**The method is meant to validate the parameter to ensure it is within the range of index values for the arraylist. A valid index must be zero or higher, and less than the size of the arraylist. This method uses the boolean operator OR instead of AND. The condition as written will always be true, resulting in an `ArrayIndexOutOfBoundsException` if the index is not valid.**