Student Performance Q&A:

2012 AP® Computer Science A Free-Response Questions

The following comments on the 2012 free-response questions for AP® Computer Science A were written by the Chief Reader, Jody Paul of Metropolitan State University of Denver. They give an overview of each free-response question and of how students performed on the question, including typical student errors. General comments regarding the skills and content that students frequently have the most problems with are included. Some suggestions for improving student performance in these areas are also provided. Teachers are encouraged to attend a College Board workshop to learn strategies for improving student performance in specific areas.

Question 1

What was the intent of this question?

This question involved object construction, list access and modification, string comparison, and implications of design choices. Students were asked to implement the method `addClimb` using two different specifications of its behavior. In both cases, the main action of `addClimb` was to use its parameters to create a new object of type `ClimbInfo` and add that object to the instance variable `climbList`. In part (a) students were required to implement `addClimb` such that the new object was simply appended to `climbList`. This could be accomplished by invoking the one-parameter `add` method of the `List` interface. In part (b) students were required to implement `addClimb` such that the new object was inserted into `climbList` so as to maintain alphabetical order of the elements. This could be accomplished by searching the list to determine the location at which the element should be inserted and invoking the two-parameter `add` method of the `List` interface. Part (c) provided an implementation of the method `distinctPeakNames` that traverses `climbList` and attempts to determine the number of distinct values of names of the peaks. Students were asked whether that implementation of `distinctPeakNames` works as intended when `addClimb` stored the objects as in part (a) and in part (b).

How well did students perform on this question?

The mean score was 4.12 out of a possible 9 points, with a standard deviation of 3.18. This question appears to have been slightly more difficult than the others.

What were common student errors or omissions?

Common student errors were associated with instance variable access, string comparison, and list modification. Many students neglected to invoke the `ClimbInfo` accessor method `getName`, attempting instead to compare a `ClimbInfo` object directly to a `String` object. When comparing two `String` objects to determine their relative order alphabetically, students inappropriately attempted to use the `equals` method or used the results of the `compareTo` method improperly. Students attempted to

insert an entry into an `ArrayList` from within an enhanced `for` loop but without exiting the loop, which would result in generating a `ConcurrentModificationException`. Another common problem was erroneous multiple insertions resulting from not exiting a loop after insertion. Other common errors involved inability to handle boundary cases, such as when `climbList` had no elements (size zero) or when the new object needed to be inserted at the end of the list in part (b).

Based on your experience of student responses at the AP Reading, what message would you like to send to teachers that might help them to improve the performance of their students on the exam?

Students need to have extensive practice with use of the `List` interface and `ArrayList` objects, especially list modification via insertion and deletion. List and array list concepts are likely to continue to constitute a substantial portion of future exams; thus students need proficiency with their use. In addition, students should be reminded to use accessor methods when provided by a class. Students would benefit from examples of and practice with different types of loops to understand when each is appropriate and how each is used. For example, the enhanced `for` loop is usually not appropriate for use with a list that is to be modified in the context of the loop. Students would also benefit from more practice using the `compareTo` method.

Question 2

What was the intent of this question?

This question involved reasoning in the context of the GridWorld case study and the design of a class using inheritance, method overriding, and instance variables to maintain state. The problem tested students' knowledge of the `Bug` class, and both creating and overriding appropriate methods. Students were required to create the `RetroBug` class, as a subclass of `Bug`, whose behavior included remembering previous direction and location, as well as the means to restore previously remembered values. Students had to override an appropriate method of the `Bug` class and write a new method. To be successful in this problem, students needed to understand a bug's behavior and the intended behavior of a retrobug, declare instance variables to remember previous state, override the `act` method, and write a `restore` method.

How well did students perform on this question?

The mean score was 4.26 out of a possible 9 points, with a standard deviation of 3.07. This question appears to have been of average difficulty.

What were common student errors or omissions?

A common error was not creating instance variables necessary to maintain the state information. Many students neglected to specify the instance of a class when attempting to invoke an instance method (for example, attempting to invoke the `isValid` method without reference to a grid object). Students appeared to lack understanding of the use of `null`, for example, attempting to assign the `null` literal to variables of primitive type or to compare primitive values to `null`. There were significant problems with writing compound conditional expressions. Students also appeared to be confused by the notion that methods other than `act` could be invoked from outside the `Bug` or `RetroBug` classes themselves.

Based on your experience of student responses at the AP Reading, what message would you like to send to teachers that might help them to improve the performance of their students on the exam?

Questions that require writing an entire class should be expected, and students need proficiency with both building a class from scratch and extending an existing class. For case study questions, students should use the reference guide to verify method signatures and the behaviors of those methods. Students can also use the reference guide for examples of class definitions. Students need to be more familiar with using grid objects and the methods used to retrieve information about the grid, locations in the grid, and actors at locations in the grid. Students should have more experience with problems that involve extending various GridWorld classes, emphasizing the principle of overriding the minimum number of methods required for a correct solution. Teachers should help students grasp the difference between variables that store primitive data types and those that store references to objects and understand the meaning and valid uses of `null`.

Question 3

What was the intent of this question?

This question entailed understanding array operations, array traversals, loops, and interfaces. Students were provided with an interface, `Horse`, and the framework for the `HorseBarn` class. In part (a) students were required to implement the method `findHorseSpace` that searches an array instance variable and returns the index of an element of the array that refers to an object whose name matches a parameter value. This could be done by traversing the array, skipping over `null` elements, and comparing the parameter with the value returned by the `getName` method of each object referenced by the elements. If no matching object was found, the method needed to return -1. In part (b) students were required to move the elements of the array such that the order of non-`null` elements was maintained, and those elements were clustered beginning at index 0 with no interspersed `null` elements. There were many ways to achieve this result, including two common approaches of either shifting array elements in place or using a temporary auxiliary collection.

How well did students perform on this question?

The mean score was 4.25 out of a possible 9 points, with a standard deviation of 3.06. This question appears to have been of average difficulty.

What were common student errors or omissions?

A large number of students did not check for `null` elements in part (a). Many students provided solutions that had array index out-of-bounds errors. Students also appeared to have great difficulty with correctly implementing the logic involved in consolidating the array elements. As in previous years, quite a number of students appeared to confuse arrays and lists.

Based on your experience of student responses at the AP Reading, what message would you like to send to teachers that might help them to improve the performance of their students on the exam?

Because array and list concepts are likely to continue to constitute a substantial portion of future exams, students need proficiency with their use. Students should have experience with algorithms that manipulate arrays and lists. They need to use the correct syntax when referencing array elements and list elements. Syntax confusion in element access may be penalized in the future even though there were no such penalties in 2012. Students should be reminded of the need to check a reference to see if it is `null`

before attempting to dereference it. Teachers could help students by providing practice with developing and implementing the logic to solve similar questions.

Question 4

What was the intent of this question?

This question addressed the use of a rectangular two-dimensional array of primitive values, accessing and modifying array elements, comparing array elements to other values, managing a counter, and understanding the distinction between void methods and those that return a value. Students were asked to implement two methods of the `GrayImage` class. In part (a) students were required to implement the method `countWhitePixels`, which determined and returned the number of array elements that contained the value 255, which was also represented by the symbolic constant `WHITE`. This could be accomplished by traversing the entire array, using nested loops, comparing every element with `WHITE`, and incrementing a counter. In part (b) students were required to implement the method `processImage`, which modified array elements on the basis of the values of elements at other indices and assigned the value 0 (symbolic constant `BLACK`) to elements whose resulting values would be less than 0. Here again a viable approach to the solution involved the use of nested loops, taking care to avoid out-of-bounds indices. (Although the question specified processing in row-major order, that aspect of the solution was not assessed because correct results were achievable via other means.)

How well did students perform on this question?

The mean score was 5.17 out of a possible 9 points, with a standard deviation of 3.23. This question appears to have been somewhat less difficult than the others.

What were common student errors or omissions?

Common errors that appeared in both part (a) and part (b) included treating the array as one-dimensional or square (both dimensions the same). Many solutions exhibited out-of-bounds errors. A significant number of students erroneously attempted to use methods of the GridWorld Grid interface on arrays, such as `getNumRows` and `getNumCols`. Similar to question 3, many students exhibited confusion between arrays and lists. In part (b) students often neglected to address both the required check for negative values and the associated assignment to `BLACK`.

Based on your experience of student responses at the AP Reading, what message would you like to send to teachers that might help them to improve the performance of their students on the exam?

Students need to exercise greater care with loop boundaries to avoid out-of-bounds errors. They should have more experience working with two-dimensional arrays and be reminded of the difference between arrays, lists, and grids. Special attention should be given to the means for determining the correct length of each of the dimensions of a rectangular array, emphasizing that not all two-dimensional arrays are square. Teachers should provide students with ample practice interpreting exam questions to ensure that students address all required aspects of a solution.