Student Performance Q&A:

2013 AP® Computer Science A Free-Response Questions

The following comments on the 2013 free-response questions for AP® Computer Science A were written by the Chief Reader, Elizabeth Johnson of Xavier University of Cincinnati, Cincinnati, Ohio. 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, and design and use of a helper method to search a list. In part (a), students were required to implement the method `getDownloadInfo` to search for an entry in the list instance variable having a title matching the parameter. This could be accomplished by traversing the list and using a string comparison to compare the parameter with the result of invoking `getTitle` on each entry. When a match was found, a reference to the object was to be returned. If the traversal completed with no match found, `null` was to be returned. In part (b), the students were required to implement `updateDownloads`, which modified entries in the list instance variable (and added to the list) based on a list parameter containing titles. Students were required to use `getDownloadInfo` in implementing this method. For each title in the parameter list, `getDownloadInfo` would be called with the title as argument. If the result was a reference to a matching `DownloadInfo` object, `incrementTimesDownloaded` would be called on that object. If the result was `null`, a new `DownloadInfo` object would be created for that title. This new object would be added to the end of the list instance variable. The result of this processing would be that the list instance variable would have been updated to include information about each title in the parameter list. Either the existing download count was incremented by one for each occurrence in the title list or, for a new title, a new representation was created with the count set to 1.

How well did students perform on this question?

This question appears to have been slightly less difficult than the others. The mean score was 4.4 out of a possible 9 points with a standard deviation of 3.35.

What were common student errors or omissions?

Some students failed to use the helper method from part (a) in implementing the part (b) method. Students used the `==` operator when the `equals` method was necessary. Students compared the entire object rather than the information required (e.g., used `DownloadInfo` versus the result of `getTitle`

invoked on the object). Students displayed a misunderstanding of how to maintain a correct count in the `DownloadInfo` objects in the list. Some students also made errors in constructing objects or did not construct a new object before inserting into the list.

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 could benefit from practice in searching a list for objects meeting certain criteria. Use of helper methods to accomplish parts of algorithms is an important skill for students to develop. Students should be encouraged to read requirements carefully before starting to write code. Students should also be encouraged to use enhanced `for` (for-each) loops where appropriate so as to avoid loop bounds errors and simplify code.

Question 2

What was the intent of this question?

This question involved use of the array data structure, array traversal, and both access and modification of array elements. Students were asked to implement a constructor and a method of the `TokenPass` class. In part (a), students were required to implement the constructor, which creates an `int` array of length `playerCount` (the constructor parameter) and initializes each element in that array to a random integer between1 and 10, inclusive. In addition, the `currentPlayer` must be initialized to a random variable between 0 and `playerCount`. In part (b), students were required to implement the method `distributeTokens`, which distributes the tokens at `board[currentPlayer]`, one at a time, to subsequent positions in the array. If tokens remain to be distributed after the last board position is updated, distribution must continue at position 0. Distribution stops when `board[currentPlayer]` tokens have been distributed.

How well did students perform on this question?

This question appears to be of average difficulty. The mean score was 4.16 out of a possible 9 points with a standard deviation of 3.06.

What were common student errors or omissions?

Many students made mistakes in the use of `Math.random` to obtain numbers within a given range. Many students provided solutions that suffered from array index out-of-bounds errors. Students made errors in logic in terms of detecting when the token distribution was completed and accounting for the situation where the end of the array was reached and tokens remained to be distributed. Students also confused arrays with arraylists. Incorrect syntax was sometimes used for array instantiation.

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 should become familiar with the Quick Reference Guide and use it during the exam to check method signatures and usage. Students would benefit from practice with use of random number generators in various contexts. Practice with array instantiation and initialization would be helpful. The importance of correct maintenance of a loop control variable should be emphasized. Examples that involve multiple traversals of an array should be presented.

Question 3

What was the intent of this question?

This question involved reasoning in the context of the GridWorld case study. Part (a) required writing a `static` method in a utilities class, traversing a two-dimensional data structure included in a `Grid`, working with a list (instantiating an `ArrayList` of `Location` objects, adding elements and testing for empty), and returning values from a method. Part (b) required the writing of a `Critter` subclass, understanding inheritance and polymorphism, overriding selected methods of the `Critter` class, and paying attention to specific post-conditions.

Students commonly approached part (a) in either of two ways.

(1) Start with an empty `ArrayList` and add empty locations.
(2) Start with an `ArrayList` of all locations and remove occupied locations.

In part (b), students needed a good understanding of GridWorld to determine which two methods (`getMoveLocations` and `selectMoveLocation`) to override. Overriding `makeMove` instead of `selectMoveLocation` violates `makeMove`'s post-condition that `getLocation() == loc` in the case `loc` is `null`.

How well did students perform on this question?

This question appears to be slightly more difficult than the others. The mean score was 4.01 out of a possible 9 points with a standard deviation of 3.31.

What were common student errors or omissions?

Common student errors were inappropriate use of the enhanced `for` (for-each) loop when removing elements from a list, failure to test for `null` before invoking a method, and overriding the wrong methods to achieve a behavior. Students also had difficulty in using the static method written in part (a) while implementing the class in part (b). Incorrectly formed tests for identifying empty locations were also common.

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?

Practice in writing and use of static methods other than Java library methods would aid student understanding. Students should have a clearer understanding of the error in using an enhanced `for` (for-each) loop when elements are removed from a list. Examples of uses (and misuses) of alternative loop types for various tasks would help in this area. 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 utilize 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. The implications of decisions about which methods to override in a subclass would also be a useful topic.

Question 4

What was the intent of this question?

This question addressed the construction, initialization and use of a rectangular two-dimensional array of primitive values, accessing array elements, managing an accumulator, and returning a value. Students were asked to implement a constructor and method of the `SkyView` class. In part (a) students were required to implement a constructor, which created a rectangular array (dimensions determined by parameters `numRows` and `numCols`) and initialized the instance array with values from a 1D parameter array. The mapping of the 1D parameter array into the 2D array was defined as alternating the direction of the fill of each row so that the first row must fill from left to right and the second from left to right and so on. In part (b) students were required to implement the method `computeAverage`, which used parameters `startRow`, `endRow`, `startCol`, `endCol` to define a rectangular region over which to compute the average. The method must create a `double` accumulator initialized to 0.0 and then traverse the relevant section of the 2D array, using an outer loop starting at `startRow`, and ending at `endRow`, and an inner loop starting at `startCol` and ending at `endCol`. Within the loops, each element is added to the sum. After the loops complete, the sum is divided by the number of elements to obtain the average. The average is returned by the method.

How well did students perform on this question?

This question appears to be of average difficulty. The mean score was 4.19 out of a possible 9 points with a standard deviation of 3.30.

What were common student errors or omissions?

Many students confused rows and columns in the two-dimensional array. Many students provided solutions that suffered from array index out-of-bounds errors. Students also had difficulty with correctly implementing the logic to alternate the direction of element placement in successive rows. There were also errors in iterating through a row from right-to-left rather than left-to-right. Students confused instance variables with local variables and did not correctly instantiate the two-dimensional array. Students also returned values from the constructor and failed to return from a non-`void` method.

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. The differences between integers and floating point numbers, especially in terms of arithmetic operations, would be a useful topic. Students need to read the questions more carefully to ensure that they understand the requirements of the problem.