CollegeBoard
Advanced Placement
Program

Student Performance Q&A:

2005 AP® Computer Science A Free-Response Questions

The following comments on the 2005 free-response questions for AP® Computer Science A were written by the Chief Reader, David Reed of Creighton University in Omaha, Nebraska. 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 focused on abstraction and data structure access. It involved storing and manipulating a collection of hotel reservations, given a `Reservation` class and the skeleton of a `Hotel` class for representing the collection. The `Hotel` class had two `private` data fields defined, an array of room reservations, and an `ArrayList` to serve as a waiting list when the rooms were full. In part (a) students needed to iterate over the array, searching for an empty room (i.e., an array entry that was `null`). If an empty room was found, then a reservation had to be created and assigned to that array entry. If not, then the customer had to be added to the `ArrayList`. In part (b) students were required to cancel a room reservation and move a customer from the waiting list if possible. This involved calling a method of the `Reservation` class to determine the correct room number, setting the corresponding entry in the `rooms` array to `null`, and then determining whether the `ArrayList` was empty. If not, then the first entry in the `ArrayList` had to be removed and assigned to the `rooms` array.

How well did students perform on this question?

In terms of its difficulty, this question was comparable to the first question on other Computer Science A Exams in previous years. Student performance was reasonable, with the second highest mean score on the exam (4.54 out of a possible 9 points). The distribution of scores was excellent, with a large number of 9s and 8s suggesting that strong students found this question easy. Scores between 7 and 1 were fairly evenly distributed, but the large number of 0s stood out, especially given the fact that this was the first question on the exam. Because of the question's emphasis on abstraction and data access, a student unable to deal with interacting classes and array/`ArrayList` traversal would be unlikely to receive much credit.

What were common student errors or omissions?

Weaker students made the typical array traversal errors—incorrect bounds, off-by-one errors, etc. The most common errors involved accessing the array and `ArrayList` incorrectly. Some students seemed confused by having both an array and an `ArrayList` as fields; they would use the incorrect form of access (either `[]` on an `ArrayList` or `.get()` on an array). Other syntactic errors included incorrectly accessing the first entry in the `ArrayList` and using `"new"` incorrectly when attempting to create a new `Reservation` object.

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?

Familiarize students with both arrays and `ArrayLists`, as both are likely to be seen on future exams. In particular, arrays are commonly used when the size of a list is fixed, while an `ArrayList` is commonly used when the list size is dynamic. Students must be careful to recognize these two different structures and use the correct form of access for each. They must also be comfortable with interacting classes, including examples such as this one, where a simple class is provided and another class stores and manipulates objects of that simple class. In addition, the mechanics of creating objects using `"new"` should be well understood.

Question 2

What was the intent of this question?

The question focused on students' ability to design a hierarchy of classes using inheritance. An `abstract Ticket` class was provided, and students were asked to design classes derived from `Ticket` that provided specialized functionality. In part (a) the students were required to design and implement a complete class representing `Advance` sales tickets. This involved declaring a `private` field for storing either the price or the number of days in advance that the ticket was purchased. It also involved writing a constructor to initialize that field and overriding the `abstract getPrice` method of the parent class. In part (b) students were required to design and implement a `StudentAdvance` ticket class, which was derived from `Advance` and gave a special discount for students. This involved writing a constructor and overriding both the `getPrice` and `toString` methods. Since data fields in `Ticket` and `Advance` were `private`, it was necessary to use `super` in the constructor and both methods in order to include the functionality of the parent methods.

How well did students perform on this question?

Despite the similarities between this question and the design question from the 2004 exam, overall performance in 2005 was not as strong. The mean score was the lowest on the exam: 4.04 out of a possible 9 points. The number of 9s was comparable to 9s scored on other questions on the exam, suggesting that the best students did not have trouble with this question. Scores between 8 and 1 were fairly evenly distributed, but the number of 1s was much higher than on other questions, and the number of 0s was only higher on question 1. This suggests that weaker students were not fully capable of designing a class from scratch and correctly utilizing inheritance. Overall, the mean score on this question was the lowest on the exam.

What were common student errors or omissions?

The poor performance of weaker students clearly shows that there is still much confusion over the concept of inheritance. Students often did not extend the proper class or used the reserved word `"implements"` rather than `"extends"`; in some cases they did not realize that connecting the classes was even necessary. Although the *AP Computer Science Course Description* speaks to the importance of a class having `private` fields for the aspect of information hiding, students defined fields that were `public` and `protected`, and some fields even had no declared type. Not recognizing that the `private` fields from parent classes were inaccessible, some students tried to access them directly instead of calling the appropriate methods. For example, the most common mistakes were in the `toString` method, where students would attempt to access the `private serialNumber` field, or call the `private getNextSerialNumber` method as opposed to calling `super.toString().`

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?

In general, students need to be more comfortable with inheritance and polymorphism. They need to see examples of inheritance hierarchies and be able to recognize when fields and methods can be inherited and when they need to be overridden. In particular, students should be aware of the limitations of `private` fields when implementing inheritance, and the use of `super` to call methods from a parent class. Design questions such as this one, which require students to make choices in the creation of classes or data structures, are likely to appear on future exams. Students must be comfortable with designing classes from scratch.

Question 3

What was the intent of this question?

The question was based on the Marine Biology Simulation Case Study and focused on abstraction and inheritance. Students needed to show their understanding of the case study and its interacting classes by writing member functions for a new `ZigZagFish` class. In part (a) students were required to override the `nextLocation` method, which selected the next location following a zig-zag pattern. The implementation of this method required students to utilize `Fish` methods to obtain the environment, location, and direction of the fish, and `Environment` methods to obtain the correct diagonal location and check to see if it was empty. In part (b) students were required to override the `move` method to produce the appropriate movement. This involved calling the `nextLocation` method, testing that location, and either moving or changing direction depending on that location.

How well did students perform on this question?

Overall, student performance was good: the mean score was 4.3 out of 9 possible points. There were a large number of 9s and 8s, and scores between 7 and 0 were fairly evenly distributed. However, there were an extremely large number of blank answers. This suggests that many students did not know the case study at all and simply chose to skip the question. Of those who attempted to answer it, the scores were very good. In fact, if 0s and blank answers were ignored for all questions, this question would have the second highest mean score on the exam. The large number of 9s and 8s may be partially attributed to similarities with the existing `DarterFish` class, whose code was included in the exam appendix. Strong students may have recognized this similarity and used the provided code as the starting point for their solution.

What were common student errors or omissions?

The most common errors occurred when students were attempting to access a forward diagonal position from the fish's current position. Many students attempted to access a position by manipulating a row or column, often referring to `private` data fields from the `Environment` class. Frequently, students failed to use the `willZigRight` field, which was intended to determine which way the fish should move. In part (b) the most common error was forgetting to update the `willZigRight` field so that the next move would be in the opposite direction.

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?

Many students did not know the case study at all, but those who had worked with it understood it pretty well. Teachers who are not covering the case study, or who are relegating it to the very end of the year, need to recognize its importance. There will be a free-response question and several multiple-choice questions based on the case study on the Computer Science A Exam every year. If all students are familiar with the case study, the exam could include more in-depth questions. Emphasizing it throughout a course ensures a greater level of understanding, and the case study can serve as a framework for introducing new topics in the course.

Question 4

What was the intent of this question?

The question focused on abstraction, array traversal, and the application of basic algorithms. In part (a) students were required to calculate the average of numbers in a section of an array, given the starting and ending indices. This involved traversing the correct section of the array, summing values, and then dividing by the size of the section. In part (b) students were required to traverse the array and test consecutive items to determine whether the numbers were arranged in increasing order. In part (c) they used the methods they wrote in the previous parts to compute a conditional average, averaging only the last half of the array if the numbers were increasing. This last part focused heavily on abstraction, as code reimplementations received no credit.

How well did students perform on this question?

This was probably the easiest and most straightforward question on the exam. It had the highest mean score (5.08 out of a possible 9 points), with scores skewed heavily toward the high end. There were few 0s and a reasonably small number of blank answers, considering this was the last question.

What were common student errors or omissions?

Given the strong overall performance, most errors were fairly minor. In part (a) errors included failing to initialize the counter to zero, performing integer division, and using an incorrect denominator (`last-first` instead of `last-first+1`) in calculating the average. In part (b) the most serious error was an early return from the loop. Some students placed an `if-else` statement inside the loop, so that a value would always be returned in the first iteration. In part (c) students lost points if they did not utilize the abstractions already created. For example, some students rewrote the code for calculating the average of an array and thus lost credit.

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?

Teachers and students need to be aware that code reuse and abstraction are important concepts on the Computer Science A Exam. The following statement appears at the beginning of the exam:

> In writing solutions for each question, you may use any of the accessible methods that are listed in classes defined in that question. Writing significant amounts of code that can be replaced by a call to one of these methods may not receive full credit.

Students need to be comfortable calling methods that are provided in existing classes or else implemented by themselves in previous parts of the question.