2018

AP CollegeBoard

AP Computer Science Principles

Sample Student Responses and Scoring Commentary

Inside:

Performance Task — Create

☑ **Scoring Guideline**

☑ **Scoring Commentary**

Student Samples provided separately

AP® Computer Science Principles — Create Performance Task
2018 Scoring Guidelines and Notes

Reporting Category	Task	Scoring Criteria	Decision Rules	Scoring Notes
Row 1 **Developing a Program with a Purpose**	**VIDEO & RESPONSE 2A**	• The video demonstrates the running of at least one feature of the program submitted. **AND** • The response (audio narration or written response) identifies the purpose of the program (what the program is attempting to do).	Response earns the point if it explains the function of the program instead of identifying the purpose. Response earns the point if the illustrated feature runs, even if it does not function as intended. Response earns the point if the video includes a narration or some form of closed captioning that addresses the purpose of the program. **Do NOT award a point if any one of the following is true:** ● a video is not submitted; ● the video does not illustrate the feature mentioned in the response; or ● the video does not illustrate the running of the feature (screen shots or storyboards are not acceptable and would not be credited).	● Purpose means the intended goal or objective of the program. ● Function means how the program works.
Row 2 **Developing a Program with a Purpose**	**RESPONSE 2B**	• Describes or outlines steps used in the incremental and iterative development process to create the entire program.	**Do NOT award a point if any one of the following is true:** ● the response does not indicate iterative development; ● refinement and revision are not connected to feedback, testing, or reflection; or ● the response only describes the development at two specific points in time.	● Development processes are iterative and cyclical in nature and require students to reflect AND improve on what they have created. Examples of iterative development could include reflection, revision, testing and refining, and improvements based on feedback. ● The incremental and iterative development process does not need to be a formal method such as waterfall, top — down, bottom-up, agile, etc.
Row 3 **Developing a Program with a Purpose**	**RESPONSE 2B**	• Specifically identifies at least two program development difficulties or opportunities. **AND** • Describes how the two identified difficulties or opportunities are resolved or incorporated.	Response earns the point if it identifies two opportunities, or two difficulties, or one opportunity and one difficulty AND describes how each is resolved or incorporated. **Do NOT award a point if any one of the following is true:** ● only one distinct difficulty or opportunity in the process is identified and described; or ● the response does not describe how the difficulties or opportunities were resolved or incorporated.	
Row 4 **Applying Algorithms**	**CODE SEGMENT IN RESPONSE 2C**	• Selected code segment implements an algorithm.	**Do NOT award a point if any one of the following is true:** ● the algorithm consists of a single instruction; ● the code segment consisting of the algorithm is not included in the written responses section or is not explicitly identified in the program code section; or ● the algorithm is not explicitly identified (i.e., the entire program is selected as an algorithm, without explicitly identifying the code segment containing the algorithm).	● Algorithms are precise sequences of instructions for processes that can be executed by a computer and are implemented using programming languages. (EU 4.1) ● Algorithms make use of sequencing, selection or iteration. (EK 4.1.1A)

Reporting Category	Task	Scoring Criteria	Decision Rules	Scoring Notes
Row 5 **Applying Algorithms**	**RESPONSE 2C**	• Selected code segment implements an algorithm that uses mathematical or logical concepts. **AND** • Explains how the selected algorithm functions. **AND** • Describes what the selected algorithm does in relation to the overall purpose of the program.	The algorithm being described can utilize existing language functionality, or library calls. Response earns the point even if the algorithm was not newly developed. (i.e., a student's reimplementation of the algorithm to find the minimum value) **Do NOT award a point if any one of the following is true:** ● the selected algorithm consists of a single instruction; ● the selected algorithm consists solely of library calls to existing language functionality; ● the selected algorithm does not include mathematical or logical concepts; ● the response only describes what the selected algorithm does without explaining how it does it; ● the response does not explicitly address the program's purpose; ● the code segment consisting of the selected algorithm is not included in the written responses section or is not explicitly identified in the program code section; or ● the algorithm is not explicitly identified (i.e., the entire program is selected as an algorithm, without explicitly identifying the code segment containing the algorithm).	● See Row 4 definitions and curriculum framework alignment. ● Mathematical concepts include mathematical expressions using arithmetic operators and mathematical functions. (EK 5.5.1.D) ● Logical concepts include Boolean algebra and compound expressions. (EK 5.5.1E and 5.5.1F) ● Iteration is the repetition of part of an algorithm until a condition is met or for a specified number of times. (EK 4.1.1D) ● Selection uses a Boolean condition to determine which of two parts of an algorithm is used. (EK 4.1.1C) ● Iteration is the repetition of part of an algorithm until a condition is met or for a specified number of times. (EK 4.1.1D) ● Selection uses a Boolean condition to determine which of two parts of an algorithm is used. (EK 4.1.1C)
Row 6 **Applying Algorithms**	**RESPONSE 2C**	• Selected code segment implements an algorithm that includes at least two or more algorithms. **AND** • At least one of the included algorithms uses mathematical or logical concepts. **AND** • Explains how one of the included algorithms functions independently.	**Do NOT award a point if any one of the following is true:** ● the selected algorithm consists of a single instruction; ● the selected algorithm consists solely of library calls to existing language functionality; ● neither of the included algorithms nor the selected algorithm that includes two or more algorithms uses mathematical or logical concepts; ● the code segment consisting of the algorithm is not included in the written responses section or is not explicitly identified in the program code section; or ● the algorithm is not explicitly identified (i.e., the entire program is selected as an algorithm, without explicitly identifying the code segment containing the algorithm).	● See Row 4 and Row 5 definitions and curriculum framework alignment.
Row 7 **Applying Abstraction**	**CODE SEGMENT IN RESPONSE 2D**	• Selected code segment is a student-developed abstraction.	Responses that use existing abstractions to create a new abstraction, such as creating a list to represent a collection (e.g., a classroom, an inventory), would earn this point. **Do NOT award a point if any one of the following is true:** ● the response is an ***existing*** abstraction such as variables, existing control structures, event handlers, APIs; ● the code segment consisting of the abstraction is not included in the written responses section or is not explicitly identified in the program code section; or ● the abstraction is not explicitly identified (i.e., the entire program is selected as an abstraction, without explicitly identifying the code segment containing the abstraction).	● The following are examples of abstractions (EK 5.3.1): o Procedures o Parameters o Lists o Application program interfaces (APIs) o Libraries ● Lists and other collections can be treated as abstract data types (ADTs) in developing programs. (EK 5.5.1I)
Row 8 **Applying Abstraction**	**RESPONSE 2D**	• Explains how the selected abstraction manages the complexity of the program.	Responses should not be penalized for explanations of abstractions that are not developed by the student. **Do NOT award a point if any one of the following is true:** ● the explanation does not apply to the selected abstraction; or ● the abstraction is not explicitly identified (i.e., the entire program is selected as an abstraction, without explicitly identifying the code segment containing the abstraction).	● See Row 7 definitions and curriculum framework alignment.

AP® COMPUTER SCIENCE PRINCIPLES
2018 SCORING COMMENTARY

Create Performance Task

Overview

Programming is a collaborative and creative process that brings ideas to life through the development of software. Programs can help solve problems, enable innovations, or express personal interests. In this performance task, students developed a program of their choice. The students' development process includes iteratively designing, implementing, and testing their program. Students were strongly encouraged to work with another student in their class.

Sample: A

Row 1: 1
Row 2: 1
Row 3: 1
Row 4: 1
Row 5: 1
Row 6: 1
Row 7: 1
Row 8: 1

Row 1:
The response earned the point for this row.
The video demonstrates the running of the program, and the purpose of the program is "for the wolf to 'eat' five blue chickens before the timer runs out."

Row 2:
The response earned the point for this row.
The response describes the iterative process in developing the program: "starting with my original method and continuing to build off of that. During the process, as I would think of an element that I needed to add to the game, I would work on the code needed for that certain part of the game."

Row 3:
The response earned the point for this row.
The response identifies a difficulty: "the first part my wolfMove method." The response indicates that it is resolved "after a lot of trial and error ... able to make the wolf move up while moving forward and make sure that when it stopped moving it was above ground." The response identifies an opportunity "to add aspects to make my game better ... the game was too easy to play." The opportunity is addressed when a timer was added to the game: "there was more pressure, which made the game realistic."

Row 4:
The response earned the point for this row.
The code that is given represents an algorithm.

Create Performance Task (continued)

Row 5:
The response earned the point for this row.
The response includes an algorithm that has math/logic (while loop, height/2). The response explains how the wolfMove algorithm works: "The method moves the wolf to the blue chickens and then the chickens' opacity is set to zero." The response describes what the algorithm does with respect to the entire program. It "contains the code that tells my wolf how to move."

Row 6:
The response earned the point for this row.
The response gives an algorithm (wolfMove) that uses two algorithms (resetColors and scoreCounter). The response explains how resetColors works: "count the number of blue chickens and proceeds to use logic and call my "setrandomcolors" ... method if there are no blue chickens left." The resetColors function includes logic using if statements.

Row 7:
The response earned the point for this row.
The boxed code segment in the response represents a valid abstraction (a procedure or method).

Row 8:
The response earned the point for this row.
The response explains how the abstraction manages complexity: "I did not have to re-write the same ten if/else statements more than once" and "it ensured that I would not have to put together the large amount of code every time that I called the method."

Sample: B

Row 1: 1
Row 2: 1
Row 3: 1
Row 4: 1
Row 5: 1
Row 6: 1
Row 7: 1
Row 8: 1

Row 1:
The response earned the point for this row.
The video illustrates the running of the program, and the response states the purpose of the program is "to help users memorize terms and their definitions more effectively."

AP® COMPUTER SCIENCE PRINCIPLES
2018 SCORING COMMENTARY

Create Performance Task (continued)

Row 2:
The response earned the point for this row.
The response describes the incremental process used to create the program. The response lists these steps: "drew a flowchart, incorporated a for loop that prints blank lines, decided to improve the program using the lower() method." The iterative process occurred during the difficulty where testing was used to find a problem where the "program always calculates the user's quiz-score 1-digit higher than it actually is" and the response includes how it was resolved.

Row 3:
The response earned the point for this row.
The response describes a difficulty ("calculates the user's quiz-scores 1-digit higher than it actually is") and its resolution ("assigning points_scored to 0"). The response includes a second difficulty ("users taking the quiz can cheat off the answers displayed in the study section") and its resolution ("printing 300 blank lines between the 'study' and 'quiz' portions").

Row 4:
The response earned the point for this row.
The circled code segment provided in the response represents an algorithm.

Row 5:
The response earned the point for this row.
The response includes math/logic through the use of if statements. The response explains how it works overall ("The for-loop iterates over each item in the dictionary and asks the user to enter the term which corresponds to a given definition"). The response also describes what this algorithm does in relation to the entire program ("quizzing the user, and the three smaller algorithms (the if-else-statements) included within the algorithm determines the score and streak-points of the user and uses that information to print adequate motivational messages").

Row 6:
The response earned the point for this row. The response explains that the three if-else structures are the independent algorithms within the main algorithm. Each of these is explained and each uses logic (an if statement along with math).

Row 7:
The response earned the point for this row. The code given in the response represents an abstraction (a procedure).

Row 8:
The response earned the point for this row. The response explains how this abstraction manages complexity by "improving readability ... and making debugging easier."

AP® COMPUTER SCIENCE PRINCIPLES
2018 SCORING COMMENTARY

Create Performance Task (continued)

Sample: C

Row 1: 1
Row 2: 1
Row 3: 1
Row 4: 1
Row 5: 1
Row 6: 0
Row 7: 1
Row 8: 1

Row 1:
The response earned the point for this row.
The video illustrates the running of the program. While the response does not include the purpose of the program, it does state how the program functions, which is allowed for this row.

Row 2:
The response earned the point for this row.
The response describes the incremental process to develop the program: "setting up the background and arranging the cards before developing the Card class ... determined the coordinates ... tested the program multiple times ... confirm the accuracy of the program by running it to check if it worked ... getting them [cards] to flip."

Row 3:
The response earned the point for this row.
The response identifies a difficulty ("trouble implementing Greenfoot's mouseClicked() method") with a resolution ("replaced [it] with a call to Greenfoot's isKeyDown() method in the same location"). The response identifies a second difficulty ("needed an easier way to restart the game") and its resolution ("developed a way to press r in order to reset the background when the Gameover screen pops up").

Row 4:
The response earned the point for this row.
The code segment provided is an algorithm.

Row 5:
The response earned the point for this row.
The algorithm provided uses math/logic (if statement). The response explains how the algorithm works: " I set a variable to a random integer value, which every card is instantiated with as a parameter ... this method tests if the card's value is the same as the "x" value, then decides if it should call the setX() method or the setCheck() method." The response also describes what the purpose is for this algorithm in relation to the entire program: "The turnCard method will set the image to either a check or x."

Create Performance Task (continued)

Row 6:
The response did not earn the point for this row.
The response identifies two algorithms used by the turnCard method, but the two methods do not use math/logic. Setting a variable to true or false is not considered use of logic, just as x = 4 is not considered use of math.

Row 7:
The response earned the point for this row.
The selected code segment shows the use of a list (cardList) to create a new abstraction to represent a collection of cards.

Row 8:
The response earned the point for this row.
The response explains how the abstraction manages program complexity: "the iterative process of going through each item in the list increases code efficiency because I no longer had to add every card to the background separately ..." Use of the list helps manage complexity by allowing for the use of a loop to process the list, reducing the amount of coding as explained: "I had around 9 separate addObject statements and 9 separate calls to the card's act() method."

Sample: D

Row 1: 1
Row 2: 0
Row 3: 1
Row 4: 1
Row 5: 1
Row 6: 0
Row 7: 1
Row 8: 1

Row 1:
The response earned the point for this row.
The response states the purpose of the program as "play a fun game of bowling," and the video illustrates the purpose.

Row 2:
The response did not earn the point for this row.
The response does not describe the overall development process of the entire program. The response focuses almost entirely on the two difficulties.

AP® COMPUTER SCIENCE PRINCIPLES
2018 SCORING COMMENTARY

Create Performance Task (continued)

Row 3:
The response earned the point for this row.
The response describes two difficulties. The first difficulty is that "every time the bowling ball touched the bowling pins, they didn't disappear." This is resolved by making the code for the bowling pins broadcast messages for the bowling ball to receive. The second difficulty is that "when all the pins were down, the host wouldn't say ‘Great job!’" This is resolved by "broadcasting a message when the last pin fell down."

Row 4:
The response earned the point for this row.
The block selected represents an algorithm.

Row 5:
The response earned the point for this row.
The algorithm uses mathematical and logical concept via the score change by adding 1 in the if statement. The response explains how this function works in detail. The response also describes that this algorithm's purpose is, "sensing whether the bowling ball is touching it or not."

Row 6:
The response did not earn the point for this row.
The algorithm given does not include two or more additional algorithms that function independently.

Row 7:
The response earned the point for this row.
The block given is an abstraction (a procedure).

Row 8:
The response earned the point for this row.
The response explains how complexity is managed: "I noticed this code could be made shorter and simpler ... it ensured that every time I would look at the code, I wouldn't get confused to see the same things repeated over and over."

AP® COMPUTER SCIENCE PRINCIPLES
2018 SCORING COMMENTARY

Create Performance Task (continued)

Sample: E

Row 1: 1
Row 2: 0
Row 3: 0
Row 4: 1
Row 5: 1
Row 6: 0
Row 7: 1
Row 8: 1

Row 1:
The response earned the point for this row.
The video illustrates the running of the program, and the response states the purpose as allowing the user to "put in whatever terms and definitions they desire and study off them later on flashcards."

Row 2:
The response did not earn the point for this row.
The response does not describe the incremental or iterative process used in developing the entire program.
The response focuses on two decisions that were made in determining what would be in the program.

Row 3:
The response did not earn the point for this row.
The response identifies an opportunity as adding functionality to allow users to enter the "definition as the word was being given." This is resolved by including "another button." The difficulty identified is the decision to use flashcards over multiple choice, which is a design choice, not a program development difficulty.

Row 4:
The response earned the point for this row.
The code segment provided is an algorithm.

Row 5:
The response earned the point for this row.
The algorithm provided uses math/logic via if statements. The response explains how the algorithm works: "When the next button is clicked, it displays the label font text and then doesn't show the back text. The index also determined which flashcard that you are on, in which you keep going to the next term as the next button is clicked. Furthermore, if the index is bigger than the number of items in the list then it restarts back to 1, or the first item in the list. This is the same for the other algorithm as they both use an index." The response also describes what the purpose is for this algorithm in relation to the entire program: "allows the user to go to the next flashcard" and "the user has a study guide environment in which they can type in the necessary term to the definition and then be able to go to the next set of terms."

Create Performance Task (continued)

Row 6:
The response did not earn the point for this row.
The response briefly states that one algorithm "makes [it] so that user can type in the term as the other algorithm is displaying the definition as a flashcard." However, it is not clear where each of these algorithms is in the supplied code segment, so it is not clear if these algorithms are included in the identified algorithm.

Row 7:
The response earned the point for this row.
The supplied code block represents a student-developed abstraction, which is a procedure.

Row 8:
The response earned the point for this row.
The response gives a reason why the abstraction manages complexity: "By creating this abstraction it makes the general coding clearer and easier to read as it is already being used once."

Sample: F

Row 1: 1
Row 2: 0
Row 3: 1
Row 4: 1
Row 5: 0
Row 6: 0
Row 7: 1
Row 8: 1

Row 1:
The response earned the point for this row.
The video illustrates features of the program, and the response states the purpose of the program.

Row 2:
The response did not earn the point for this row.
The response does not describe the incremental and iterative design process for the entire program.

Row 3:
The response earned the point for this row.
The response identifies two difficulties and how they are resolved. The first difficulty is "setting the timer to countdown 30 seconds and stopping the time when the pause button was clicked." Its resolution is "a function called Time and called it in the event handler of the resume button, which in result would change just the time in intervals of 1000 milliseconds, but not move back to 30 seconds." The second difficulty is "updating the high score the user received every time the score was higher." Its resolution is "to add in an if statement, such that if the score the user received was greater than the score from other games, it would update the high score value to that score."

Create Performance Task (continued)

Row 4:
The response earned the point for this row.
The code given in the response is an algorithm because it involves sequencing, selection and/or iteration.

Row 5:
The response did not earn the point for this row.
Although the algorithm given includes math minimally (incrementing the score) and it explains how the functions work, it does not describe what this algorithm does in relation to the overall program.

Row 6:
The response did not earn the point for this row.
The response does not clearly identify two algorithms used by the selected algorithm.

Row 7:
The response earned the point for this row.
The code segment given represents an abstraction (a procedure or function).

Row 8:
The response earned the point for this row.
The response explains how the abstraction manages complexity for the program by stating: "the code is reduced and not being repeated on certain event handlers."

Sample: G

Row 1: 1
Row 2: 0
Row 3: 1
Row 4: 1
Row 5: 1
Row 6: 0
Row 7: 0
Row 8: 0

Row 1:
The response earned the point for this row.
The video illustrates the program running, and the response states the purpose of the program as shown in the video.

Row 2:
The response did not earn the point for this row.
The response does not describe the overall development of the entire program.

AP® COMPUTER SCIENCE PRINCIPLES
2018 SCORING COMMENTARY

Create Performance Task (continued)

Row 3:
The response earned the point for this row.
The response identifies two difficulties during the program development. The first difficulty is "figuring out how to get the Snake to grow" and is resolved by making "the snake create a clone every time it got an apple ... it waits a while then deletes some of the clones so the snake isn't too long." The second difficulty is "changing the speed of the snake." This is resolved by "add a certain amount of steps to the original speed of the snake every time it got an apple."

Row 4:
The response earned the point for this row.
The selected block of code represents an algorithm.

Row 5:
The response earned the point for this row.
The included code segment contains math/logic by adding the score to 1. The response explains how the algorithm works: "When the snake touches the the [sic] sprite that is the apple, the random number function makes it reapear any where [sic] between x: -230 to 230 and y: -160 to 160." The response describes what this algorithm does in relation to the program: "to make the apple appear in a new place."

Row 6:
The response did not earn the point for this row.
The response does not identify two algorithms that are used within the selected algorithm nor how one of them works.

Row 7:
The response did not earn the point for this row.
The boxed code fragment does not represent an abstraction.

Row 8:
The response did not earn the point for this row.
The response does not explain how the boxed code manages complexity in the program. Making the program "more challenging" is not the purpose of an abstraction.

AP® COMPUTER SCIENCE PRINCIPLES
2018 SCORING COMMENTARY

Create Performance Task (continued)

Sample: H

Row 1: 1
Row 2: 0
Row 3: 0
Row 4: 1
Row 5: 1
Row 6: 0
Row 7: 0
Row 8: 0

Row 1:
The response earned the point for this row.
The video shows the program execution, and the written response states the purpose of the program is "to educate people on foreign languages in a fun and engaging way."

Row 2:
The response did not earn the point for this row.
The response does not describe the incremental and iterative development process for the program. Instead, the response describes how the program functions and states the main components of the program code.

Row 3:
The response did not earn the point for this row.
The response does not identify two difficulties and/or opportunities and how they are resolved or handled. The response only identifies one difficulty, clearing the user input box.

Row 4:
The response earned the point for this row.
The response identifies a segment of code that is an algorithm.

Row 5:
The response earned the point for this row.
The response uses math and/or logic (an if statement). The response states what the algorithm does in relation to the program ("used when the user must type in a response"). The response explains how the algorithm works ("There is only one correct answer, in this case it is horse. If the user types in horse exactly then they got it right and they move on; anything other than horse will be counted as wrong and the user will have to restart.").

Row 6:
The response did not earn the point for this row.
The response identifies the event handler as the first algorithm, and the if/else statement as a second algorithm. This does not meet the requirements for this row. The response needs to identify an algorithm that uses two additional algorithms that function independently.

Create Performance Task (continued)

Row 7:
The response did not earn the point for this row.
The code fragment given is not a student-developed abstraction. The abstraction given, an event handler, is built in to this language.

Row 8:
The response did not earn the point for this row.
The response does not explain how the supplied abstraction manages complexity for the program. The response only explains how the code fragment works in relation to the entire program.

Sample: I

Row 1: 1
Row 2: 0
Row 3: 0
Row 4: 1
Row 5: 1
Row 6: 0
Row 7: 0
Row 8: 0

Row 1:
The response earned the point for this row.
The video illustrates the program running, and the response states the purpose of the program as shown in the video.

Row 2:
The response did not earn the point for this row.
The response does not describe the overall interactive or iterative development of the entire program.

Row 3:
The response did not earn the point for this row.
The response states that "One main problem was to learn how to get the app to display in the countdown slot..." However, this is not a problem or opportunity during the program development process, it's a design issue. The second difficulty and its resolution are given.

Row 4:
The response earned the point for this row.
The selected block of code represents an algorithm.

Create Performance Task (continued)

Row 5:
The response earned the point for this row.
The circled code contains math/logic. The response explains how the algorithm works in detail. The response describes what this algorithm does in relation to the program: "Both algorithms are essential because in unison they allow the program to work" AND then states what would happen to the program if either part is missing. By explaining what happens if each part is missing, the response is explaining what these algorithms do for the whole program.

Row 6:
The response did not earn the point for this row.
The response identifies two algorithms, one of which is inside the other. There is no additional algorithm identified.

Row 7:
The response did not earn the point for this row.
The boxed code fragment does not represent an abstraction.

Row 8:
The response did not earn the point for this row.
The response does not explain how the boxed code manages complexity in the program. The response merely states how the code fragment works.

Sample: J

Row 1: 1
Row 2: 0
Row 3: 0
Row 4: 0
Row 5: 0
Row 6: 0
Row 7: 0
Row 8: 0

Row 1:
The response earned the point for this row.
The video illustrates features of the program and the response states the purpose of the program.

Row 2:
The response did not earn the point for this row.
The response does not describe the incremental and iterative development of the entire program.

AP® COMPUTER SCIENCE PRINCIPLES
2018 SCORING COMMENTARY

Create Performance Task (continued)

Row 3:
The response did not earn the point for this row.
The response does not identify two difficulties and/or opportunities related to the program development process and how these programming issues were resolved or handled. The issues identified have to do with "what particular questions my partner and I would come up with" and "what genre our game would be based on."

Row 4:
The response did not earn the point for this row.
The code given in the response is not considered an algorithm because each event handler consists of a single instruction that does not involve sequencing, selection, or iteration.

Row 5:
The response did not earn the point for this row.
The provided code does not use math or logic, and the code segment is not considered an algorithm.

Row 6:
The response did not earn the point for this row.
The response does not clearly identify two additional algorithms that are used by the selected algorithm. This cannot be done because the selected code is not considered an algorithm.

Row 7:
The response did not earn the point for this row.
The code segment given does not represent a student-defined abstraction. An event handler is built-in to this language.

Row 8:
The response did not earn the point for this row.
The response does not explain how this abstraction chosen manages complexity of the program. The response states "It was complex making the game move automatically to the next question after you answer it correctly." This is not the correct meaning of complexity for this row.