
AP[®] Computer Science Principles

Sample Student Responses and Scoring Commentary Set 1

Inside:

Written Response 1

- ☒ **Scoring Guidelines**
- ☒ **Student Samples**
- ☒ **Scoring Commentary**

Digital Portfolio Components provided separately

Video, Program Requirements, and Written Response 1**3 points****General Scoring Notes**

- Written responses should be evaluated solely on the rationale provided.
- Responses must demonstrate all scoring criteria, including those within bulleted lists, in each reporting category to earn the point for that category.
- Terms and phrases defined in the terminology list are italicized when they first appear.

Reporting Category	Scoring Criteria	Decision Rules
Course Project: Video (0–1 points)	The video demonstrates the running of the program including: <ul style="list-style-type: none"> • <i>input</i> • <i>program functionality</i> • <i>output</i> 	Consider the video (or Program Code if necessary) when scoring this point. <ul style="list-style-type: none"> • The video needs to show at least one example of the program’s functionality. • If the source of the input is unclear from the video, consider the full program code file when scoring this point. Do NOT award a point if the following is true: <ul style="list-style-type: none"> • The video does not show a demonstration of the program running (screenshots or storyboards are not acceptable and would not be credited).
Course Project: Program Requirements (0–1 points)	The program code includes: <ul style="list-style-type: none"> • <i>A student-developed procedure</i> • <i>A call to the student-developed procedure</i> • <i>A list (or other collection type)</i> • <i>A use of the list</i> • <i>Selection</i> • <i>Iteration</i> 	Consider the Personalized Project Reference (or Program Code if necessary) when scoring this point. <ul style="list-style-type: none"> • If the program requirements do not appear in the Personalized Project Reference, consider the full program code file when scoring this point. • The procedure does not need to have a <i>parameter</i> to earn this point. • The <i>code segments</i> demonstrating selection and iteration do not need to appear in the same <i>algorithm</i> to earn this point. • The code segments demonstrating selection and iteration do not need to be contained in a procedure to earn this point. Do NOT award a point if any one or more of the following is true: <ul style="list-style-type: none"> • The list is a one-element list. • The use of the list is irrelevant (i.e., not connected to the program’s functionality). • The call to the procedure is inconsistent with the procedure header (unless allowed by the programming language). • The use of either the selection or the iteration is trivial (i.e., does not affect the outcome of the program).

Reporting Category	Scoring Criteria	Decision Rules
Written Response 1: Program Design, Function, and Purpose (0–1 points)	<p>The written response:</p> <ul style="list-style-type: none"> identifies at least one example output of the program. explains how the identified output(s) show an aspect of the program’s functionality. 	<p>Consider the Video and Written Response 1 when scoring this point.</p> <ul style="list-style-type: none"> If the video is not available or does not show the described output, consider the full program code file when scoring this point. Either a specific example of the output or a general description of the output can meet this criteria. The response may describe more than one output but only needs to explain how at least one of the outputs shows an aspect of the program’s functionality. If the response describes more than one output, all descriptions given must be correct. <p>Do NOT award a point if any one or more of the following is true:</p> <ul style="list-style-type: none"> The identification of the example output(s) are implausible, inaccurate, or inconsistent with the program. The explanation of how the output(s) show an aspect of the program’s functionality is implausible, inaccurate, or inconsistent with the program. The output does not arise from the normal execution of the program (e.g., an error message produced by a compiler).

AP Computer Science Principles Create Performance Task Terminology

Algorithm: An algorithm is a finite set of instructions that accomplish a specific task. Every algorithm can be constructed using combinations of sequencing, selection, and iteration.

Arguments: The values of the parameters when a procedure is called.

Code segment: A code segment refers to a collection of program statements that are part of a program. For text-based, the collection of program statements should be continuous and within the same procedure. For block-based, the collection of program statements should be contained in the same starter block or what is referred to as a “Hat” block.

Collection type: Aggregates elements in a single structure. Some examples include: databases, hash tables, dictionaries, sets, or any other type that aggregates elements in a single structure.

Data stored in a list: Input into the list can be through an initialization or through some computation on other variables or list elements.

Input: Program input is data that are sent to a computer for processing by a program. Input can come in a variety of forms, such as tactile (through touch), audible, visual, or text. An event is associated with an action and supplies input data to a program.

Iteration: Iteration is a repetitive portion of an algorithm. Iteration repeats until a given condition is met or for a specified number of times. The use of recursion is a form of iteration.

List: A list is an ordered sequence of elements. The use of lists allows multiple related items to be represented using a single variable. Lists are referred to by different terms, such as arrays or arraylists, depending on the programming language.

List being used: Using a list means the program is creating new data from existing data or accessing multiple elements in the list.

Output: Program output is any data that are sent from a program to a device. Program output can come in a variety of forms, such as tactile, audible, visual, movement, or text.

Parameter: A parameter is an input variable of a procedure. Explicit parameters are defined in the procedure header. Implicit parameters are those that are assigned in anticipation of a call to the procedure. For example, an implicit parameter can be set through interaction with a graphical user interface.

Procedure: A procedure is a named group of programming instructions that may have parameters and return values. Procedures are referred to by different names, such as method, function, or constructor, depending on the programming language. A procedure is executed through the use of a procedure call.

Program functionality: The behavior of a program during execution, often described by how a user interacts with it.

Purpose: The problem being solved or creative interest being pursued through the program.

Selection / conditional statement: A selection / conditional statement affects the sequential flow of control by executing different statements based on a condition being true or false. The use of if-statements and try / exception statements are examples of selection / conditional statements.

Sequencing: The application of each step of an algorithm in the order in which the code statements are given.

Student-developed procedure / algorithm: Program code that is student-developed has been written (individually or collaboratively) by the student who submitted the response. Calls to existing program code or libraries can be included but are not considered student-developed. Event handlers are built-in abstractions in some languages and will therefore not be considered student-developed. In some block-based programming languages, event handlers begin with “when”.

An example output of the program created would be the screen displaying a diagram of interphase in mitosis as well as a description of that stage of the cell cycle once the start button has been pressed by the user. This output demonstrates an aspect of the program in question's functionality because it displays one instance of how the program explains the stages of the cell cycle, in order, to the user to educate them on the subject. This is done via the screen displaying a now visible diagram of the stage in question, in this case interphase, as well as a variety of labels explaining the stage in depth to the user, enabling them to receive multifaceted instruction on the subject, fulfilling the program's function and demonstrating its functionality.

A output of my program is "say Hooray for 2 secs" i chose to put that as my output due to the fact that if you play the game and get it right my character will tell you Hooray but if you get that wrong it will tell you "too bad that is not correct Buddy". This all goes back to my program becuae my program is a question game you get how many tries you want then you start by puting your name or what ever name you please to put then it ask you how many times you want to play then it will explain what the game is about and finally it will tell you good luck , then the game will start depending on the amount of tries you put the game will last. Personaly if you know alot about cars especialy the Dodge brand and the motors you are going to love this game but if you dont you are going to have a hard time. Aonother output my game has is "Thanks for playing you got points out of attemps correct!" were it states points and attemps it is up to how many points you got on the game and how many times you played the game. The point are based on questions you get correct, and each question you get correct it is one point for you, if you dont get any and you play three times it will tell you "Thanks for playing you got zero points out of three attemps correct!". My game has many outputs but if you say the most importnt ones these are it the other ones dont really need that much to explain.

An output in my program is the count of clicks that gets displayed showing the total number of times the user clicks one of the buttons in the game since the count of clicks gets increased everytime a button is clicked. This shows how the clicks are recorded when playing. It is an important part that represents a score for the user to see.

In procedure 1 on line 32 we see a "setText" function that displays the end value of the "score" value. the score value is the output value of the total number of targets clicked as represented in line 25, where you press a targetButton your score increses by 1.

An example output in the program is . This output shows an aspect of the program's functionality because all the previous code must be computed in order to arrive to and properly calculate this output and provide the price for the customer.

An example of an output of my program would be an interactive video game where you would be able to interact and move the in game controllable sprite left and right with your left and right arrow keys, and also shoot arrows at the randomly moving targets from your sprite by pressing the space key on your keyboard. This shows an aspect of my program's functionality because the output correctly follows all the code written in my program without any bugs or hinderences in functionality.

Question 1

Note: Student samples are quoted verbatim and may contain spelling and grammatical errors.

Overview

NEW for 2025: The question overviews can be found in the *Chief Reader Report on Student Responses* on [AP Central](#).

Sample: A

Score:

Video: 1

Program Requirements: 1

Question 1: 1

Video:

The response earned this point, demonstrating all three criteria:

- The response demonstrates input by showing the user clicking the mouse button to advance to the next display.
- The response demonstrates program functionality by showing the user reviewing the stages of the cell cycle with drawings and textual information.
- The response demonstrates output by showing, “Incorrect! The correct answer is cytokinesis!”, in response to a review question.

Program Requirements:

The response earned this point, demonstrating all six criteria:

- The response includes a student-developed procedure: `changeStages(stage)`
- The response calls the procedure: `changeStages(app.cycleStage)`
- The response includes selection: `if (app.cyleStage == 1)`
- The response included iteration: `for index in range(len(appTriviaAnswers))`
- The response included a list or collection: `app.triviaAnswers`
- The response shows the list or collection being used in the program in part ii of the List section of the PPR.

Question 1:

The response earned this point, demonstrating both criteria:

- The response identifies an example output of the program: “the screen displaying a diagram of interphase in mitosis as well as a description of that stage of the cell cycle.”
- The response explains how the identified output shows an aspect of the program’s functionality: “it displays one instance of how the program explains the stages of the cell cycle, in order, to the user to educate them on the subject.”

Question 1 (continued)**Sample: B****Score:****Video: 1****Program Requirements: 1****Question 1: 1**

Video:

The response earned this point, demonstrating all three criteria:

- The response demonstrates input by using a keyboard to type, “3.”
- The response demonstrates program functionality by asking questions and allowing the user to enter answers.
- The response demonstrates output by showing, “Too bad that is not correct buddy”, in response to an incorrect answer.

Program Requirements:

The response earned this point, demonstrating all six criteria:

- The response includes a student-developed procedure: `games+and+tries(attempts)`
- The response calls the procedure: `games+and+tries(attempts)`
- The response included selection: `if answer = item(question) of cars`
- The response includes iteration: `repeat(attempts)`
- The response includes a list: `speed`
- The response shows the list being used in the program in part ii of the List section of the PPR.

Question 1:

The response earned this point, demonstrating both criteria:

- The response explains an example output as the message, “say Hooray for 2 secs.” The response also states an additional output: “Aonother output my game has is “Thanks for playing you got points out of attempts correct!” were it states points and attempts it is up to how many points you got on the game and how many times you played the game.”
- The response explains how the first output identified shows an aspect of the program’s functionality: “if you play the game and get it right my character will tell you Hooray but if you get that wrong it will tell you “too bad that is not correct Buddy”.”

Question 1 (continued)**Sample: C****Score:****Video: 1****Program Requirements: 0****Question 1: 1**

Video:

The response earned this point, demonstrating all three criteria:

- The response demonstrates input by showing the user clicking “Start” and “Next” buttons.
- The response demonstrates program functionality by removing dog pictures that are matched with the name.
- The response demonstrates output by displaying the pictures of the dogs and their corresponding names.

Program Requirements:

The response did not earn this point, demonstrating five of the six criteria:

- The response includes a student-developed procedure: `clicked()`
- The response calls the procedure: `clicked()`
- The response includes selection: `if (im == image[1] && im2 == Name[1])`
- The response does not include iteration.
- The response includes a list or collection: `Image`
- The response shows the list or collection being used in the Program Code.

Question 1:

The response earned this point, demonstrating both criteria:

- The response identifies the output as “the count of clicks that gets displayed.”
- The response explains how the output shows an aspect of the program’s functionality: “showing the total number of times the user clicks one of the buttons of the game, since the count of clicks gets increased everytime a button is clicked.”

Question 1 (continued)**Sample: D****Score:****Video: 1****Program Requirements: 0****Question 1: 1**

Video:

The response earned this point, demonstrating all three criteria:

- The response demonstrates input by clicking the moving squares.
- The response demonstrates program functionality by moving the square after each time a click occurs to test the player's aim.
- The response demonstrates output by displaying the final score after the game ends.

Program Requirements:

The response did not earn this point, demonstrating four out of six criteria:

- The response includes a student-developed procedure: `startGame()`
- The response calls the procedure: `startGame()`
- The response includes selection: `if (timeLeft <= 0)`
- The response includes iteration: `timedLoop(1000, function())`
- The response does not include a list or collection.
- The response does not show the list or collection being used in the program in the PPR or Program Code.

Question 1:

The response earned this point, demonstrating both criteria:

- The response identifies the output as “the score value.”
- The response explains how the output demonstrates program functionality by stating that the output is “the total number of targets clicked..., where you press a targetButton your score increases by 1.”

Question 1 (continued)**Sample: E****Score:****Video: 1****Program Requirements: 0****Question 1: 0**

Video:

The response earned this point, demonstrating all three of the three criteria:

- The response demonstrates input by showing the “Next” button being pressed.
- The response demonstrates program functionality by showing ice cream flavors in response to the “Next” button being pressed.
- The response demonstrates output by showing the screen with the text: “Your order has been made...”

Program Requirements:

The response did not earn this point, demonstrating none of the six criteria:

- The response does not include a student-developed procedure.
- The response does not include a call to a student-developed procedure.
- The response does not include selection.
- The response does not include iteration.
- The response does not include a list or collection.
- The response does not show a list or collection being used.

Question 1:

The response did not earn this point, demonstrating neither of the two criteria:

- The response does not identify an example output of the program. The response states, “An example output in the program is .” and then refers to the output as “this output” for the rest of the response.
- The response explains how the output shows an aspect of the program’s functionality by stating, “all the previous code must be computed in order to arrive to and properly calculate this output and provide the price for the customer,” but this description is inconsistent with the program, which does not calculate or provide the price for the customer.

Question 1 (continued)**Sample: F****Score:****Video: 1****Program Requirements: 0****Question 1: 0**

Video:

The response earned this point, demonstrating all three criteria:

- The response demonstrates input by using the keyboard to move and shoot.
- The response demonstrates program functionality by showing the arrows moving in response to the keyboard presses.
- The response demonstrates output by showing change of position of the arrow in response to keyboard presses.

Program Requirements:

The response did not earn this point, demonstrating five of the six criteria:

- The response includes a student-developed procedure: `onStep()`
- The response does not call the procedure.
- The response includes selection: `if bullet.hitsShape (shape)`
- The response includes iteration: `for bullet in bullets`
- The response includes a list: `bullets`
- The response shows the list being used in the program in part ii of the List section of the PPR.

Question 1:

The response did not earn this point, demonstrating neither of the criteria:

- The response identifies “an interactive game” as the output. However, this is not an output of the program.
- The response does not explain how the identified output shows an aspect of the program’s functionality because the response incorrectly identifies the output.