

2024



AP[®] Computer Science Principles

Scoring Guidelines Set 1

© 2024 College Board. College Board, Advanced Placement, AP, AP Central, and the acorn logo are registered trademarks of College Board. Visit College Board on the web: collegeboard.org.

AP Central is the official online home for the AP Program: apcentral.collegeboard.org.

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"> • A <i>student-developed procedure</i> • A call to the student-developed procedure • A <i>list</i> (or other <i>collection type</i>) • A use of the list • <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"> describes at least one valid input to the program. describes what the program does with the input. 	<p>Consider the Video (or Program Code if necessary) 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 input, consider the full program code file when scoring this point. Either a specific example of the input or a description of the input can meet this criteria. If a response describes an interaction with a device (e.g., mouse or keyboard) as input, it must describe what the program does with the input resulting from the interaction. Other forms of input could include databases, device inputs such as sensors, or command line arguments. The response does not need to describe all valid inputs to the program and what the program does with all of these inputs, but all descriptions given must be correct. <p>Do NOT award a point if the following is true:</p> <ul style="list-style-type: none"> The description of the valid input is implausible, inaccurate, or inconsistent with the program. The description of what the program does with the input(s) is implausible, inaccurate, or inconsistent with the program.

Written Response 2

3 points

- General Scoring Notes**
- Written responses should be evaluated solely on the rationale provided.
 - Written 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
Written Response 2(a): Algorithm Development (0–1 points)	The written response: <ul style="list-style-type: none">• describes what is being accomplished by the code in the body of the <i>iteration</i> statement.	<p>Consider the Personalized Project Reference and Written Response 2(a) when scoring this point.</p> <ul style="list-style-type: none">• If multiple iteration statements are included in the Procedure section of the Personalized Project Reference, use the first iteration statement to determine whether the point is earned.• The first iteration statement can be found in either part (i) or part (ii) of the Procedure section of the Personalized Project Reference.• The iteration statement does not need to be contained in a procedure to earn this point.• If a procedure is identified, it does not need to contain a parameter to earn this point.• The response may describe a summary of what the iteration does in the context of the program or describe the purpose of each statement in the body of the iteration. <p>Do NOT award a point if any one or more of the following is true:</p> <ul style="list-style-type: none">• The Procedure section of the Personalized Project Reference does not contain an iteration statement.• The description of what is being accomplished by the code does not match the code in the body of the first iteration statement.• The response only restates the lines of code in the body of the iteration statement.• The response describes a trivial use of iteration.• The response describes an iteration statement or behavior that is implausible, inaccurate, or inconsistent with the program.

Reporting Category	Scoring Criteria	Decision Rules
<p>Written Response 2(b): Errors and Testing</p> <p>(0–1 points)</p>	<p>The written response:</p> <ul style="list-style-type: none"> includes two calls to the <i>procedure</i>. Each call must cause a different program <i>code segment</i> in the procedure to execute. describes the expected behavior of each call. <p>OR</p> <ul style="list-style-type: none"> explains why it is not possible for two calls to the procedure to cause different code segments to execute. 	<p>Consider the Personalized Project Reference and Written Response 2(b) when scoring this point.</p> <ul style="list-style-type: none"> If multiple procedures are included in part (i) of the Procedure section of the Personalized Project Reference: <ul style="list-style-type: none"> Use the procedure identified in the written response to determine whether the point is earned. If no procedure is identified in the written response, then use the first procedure to determine whether the point is earned. The parameter(s) used in the procedure must be explicit. Explicit parameters are defined in the header of the procedure. A procedure that uses its parameter(s) to execute two different code segments can earn this point. A procedure that uses its parameter(s) to execute or bypass a code segment can earn this point. The syntax of the procedure calls does not need to be correct as long as the correct arguments are identified. A description of each call rather than program code is acceptable. A general description of argument value(s) is considered acceptable. <p>Do NOT award a point if any one or more of the following is true:</p> <ul style="list-style-type: none"> A procedure is not identified in part (i) of the Procedure section of the Personalized Project Reference. The response does not apply to the procedure in part (i) of the Procedure section of the Personalized Project Reference. The procedure identified in part (i) of the Procedure section of the Personalized Project Reference does not include at least one explicit parameter. The use of the explicit parameter is irrelevant (e.g., does not affect the code segment of the procedure that is executed or is reassigned before being used). The two calls are to two different procedures. The response describes expected behavior that is implausible, inaccurate, or inconsistent with the program.

Reporting Category	Scoring Criteria	Decision Rules
Written Response 2(c): Data and Procedural Abstraction (0–1 points)	<p>The written response:</p> <ul style="list-style-type: none"> explains in detailed steps an <i>algorithm</i> that uses <code>checkValidity</code> to check whether all elements in the <i>list</i> are considered valid. 	<p>Consider the Personalized Project Reference and Written Response 2(c) when scoring this point.</p> <ul style="list-style-type: none"> If multiple lists are included in the List section of the Personalized Project Reference, use the list identified in the written response to determine whether the point is earned. The algorithm can be described in code, pseudocode, as a sequence of steps in English, or as a paragraph in English. The algorithm must describe how each element of the identified list is passed into <code>checkValidity</code> at least up to the first invalid element, if applicable. <p>Do NOT award a point if any one or more of the following is true:</p> <ul style="list-style-type: none"> A list is not identified in the List section of the Personalized Project Reference. If the algorithm described assumes the list contains a single element. The list identified in the Personalized Project Reference is not referenced in the response. The response implements <code>checkValidity</code> rather than describing its use. The response is too vague to allow another programmer to recreate the algorithm.

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.”