AP® Computer Science Principles

About the Advanced Placement Program® (AP®)

The Advanced Placement Program® has enabled millions of students to take college-level courses and earn college credit, advanced placement, or both, while still in high school. AP Exams are given each year in May. Students who earn a qualifying score on an AP Exam are typically eligible, in college, to receive credit, placement into advanced courses, or both. Every aspect of AP course and exam development is the result of collaboration between AP teachers and college faculty. They work together to develop AP courses and exams, set scoring standards, and score the exams. College faculty review every AP teacher's course syllabus

AP Computer Science Program

There are two computer science offerings, and students can take either course in any order or concurrently:

- AP Computer Science A focuses on computing skills related to programming in Java.
- AP Computer Science Principles complements AP Computer Science A as it aims to broaden participation in the study of computer science.

The courses underscore the importance of communicating solutions appropriately and in ways that are relevant to current societal needs. AP Computer Science courses can help address traditional issues of equity, access, and broadening participation in computing while providing a strong and engaging introduction to fundamental areas of the discipline.

AP Computer Science Principles Course Overview

AP Computer Science Principles introduces students to the breadth of the field of computer science. In this course, students will learn to design and evaluate solutions and to apply computer science to solve problems through the development of algorithms and programs. They will incorporate abstraction into programs and use data to discover new knowledge. Students will also explain how computing innovations and computing systems, including the Internet, work, explore their potential impacts, and contribute to a computing culture that is collaborative and ethical.

PREREQUISITES

It is recommended that students in the AP Computer Science Principles course have successfully completed a first-year high school algebra course with a strong foundation of basic linear functions, composition of functions, and problem-solving strategies that require multiple approaches and collaborative efforts. In addition, students should be able to use a Cartesian (x, y) coordinate system to represent points on a plane. It is important that students and their advisers understand that any significant computer science course builds upon a foundation of mathematical reasoning that should be acquired before attempting such a course. Prior computer science experience is not required to take this course.

COMPUTER LANGUAGE

AP Computer Science Principles does not have a designated programming language. Teachers have the flexibility to choose a programming language(s) that is most appropriate for their students to use in the classroom.

AP Computer Science Principles Course Content

The following are the major areas of study, or big ideas, that serve as the foundation of the course, enabling students to create meaningful connections among concepts and develop deeper conceptual understanding:

- **Creative Development:** When developing computing innovations, developers can use a formal, iterative design process or a less rigid process of experimentation, and will encounter phases of investigating and reflecting, designing, prototyping, and testing. Collaboration is an important tool at any phase of development.
- **Data:** Data are central to computing innovations because they communicate initial conditions to programs and represent new knowledge.
- **Algorithms and Programming:** Programmers integrate algorithms and abstraction to create programs for creative purposes and to solve problems.
- **Computing Systems and Networks:** Computer systems and networks are used to transfer data.
- **Impact of Computing:** Computers and computing have revolutionized our lives. To use computing safely and responsibly, we need to be aware of privacy, security, and ethical issues.

Each big idea is broken down into teachable segments called topics.

AP Computer Science Principles Computational Thinking Practices

The following computational thinking practices describe what skills students should develop during the course:

- **Computational Solution Design:** Design and evaluate computational solutions for a purpose.
- **Algorithms and Program Development:** Develop and implement algorithms.
- **Abstraction in Program Development:** Develop programs that incorporate abstractions.
- **Code Analysis:** Evaluate and test algorithms and programs.
- **Computing Innovations:** Investigate computing innovations.
- **Responsible Computing:** Contribute to an inclusive, safe, collaborative, and ethical computing culture.

AP Computer Science Principles Exam Structure

AP COMPUTER SCIENCE PRINCIPLES EXAM: 2 HOURS

Assessment Overview

The AP Computer Science Principles Exam assesses student understanding of the computational thinking practices and learning objectives outlined in the course framework. The exam consists of the Create performance task and an end-of-course AP Exam. The Create performance task requires at least 12 hours of dedicated class time for students to complete. The end-of-course exam is 2 hours long and includes 70 multiple-choice questions.

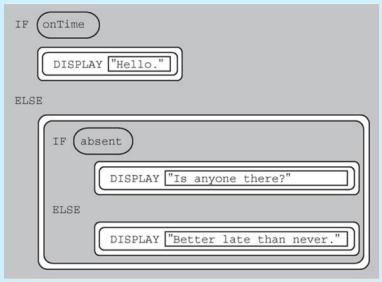
Format of Assessment

Section I:	Multiple-choice | 70 Questions | 2 Hours | 70% of Exam Score
	• 57 single-select multiple-choice questions. • 5 single-select multiple-choice questions with reading passage about a computing innovation. • 8 multi-select multiple-choice questions.
Section II:	Create Performance Task | 1 Question | At least 12 hours of class | 30% of Exam Score
	• Through-course performance task • Assesses Computational Thinking Practices 1, 2, 3, and 4

Exam Components

Sample Multiple-Choice Question

Consider the code segment below.



If the variables `onTime` and `absent` both have the value `false`, what is displayed as a result of running the code segment?

(A) `Is anyone there?`

(B) `Better late than never.`

(C) `Hello. Is anyone there?`

(D) `Hello. Better late than never.`

Correct Answer: B

Exam Components *Continued*

Performance Task: Create – Applications from Ideas

This performance task focuses on students developing computer programs and describing significant aspects of the program that allow it to run correctly.

- Students have the flexibility to write programs that reflect their interests (e.g., their desire to solve a problem; program a game; or produce digital art appealing to a specific audience, etc.). This allows students to engage in the study of computer science from a creative perspective. Students will provide evidence of their knowledge of important programming concepts such as developing algorithms and using abstractions. Students are required to submit an individual program but are able to collaborate on the development of their program.

Performance Task: Explore – Impacts of Computing Innovations

This performance task focuses on students using and applying computational analysis in the exploration of a significant computing innovation to determine and describe the impact of the innovation on people and society.

- Students select and explore an innovation of their choosing. Then, they create a computational artifact about the innovation and describe how it works and how it is used, its purpose, how it consumes and/or produces data, and the harmful and beneficial effects of the innovation on people and society.

© 2026 The College Board. APD-959