

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 AP computer science offerings, and students can take either course in any order. The AP Computer Science A course and exam continues to focus on computing skills related to programming in Java. The new AP Computer Science Principles course 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 A Course Overview

AP Computer Science A is equivalent to a first-semester, college-level course in computer science. The course introduces students to computer science with fundamental topics that include problem solving, design strategies and methodologies, organization of data (data structures), approaches to processing data (algorithms), analysis of potential solutions, and the ethical and social implications of computing. The course emphasizes both object-oriented and imperative problem solving and design using Java language. These techniques represent proven approaches for developing solutions that can scale up from small, simple problems to large, complex problems. The AP Computer Science A course curriculum is compatible with many CS1 courses in colleges and universities.

RECOMMENDED PREREQUISITES

The assumed prerequisites for entering the AP Computer Science A course include knowledge of basic English and algebra. A student in the AP Computer Science A course should be comfortable with functions and the concepts found in the uses of function notation, such as $f(x) = x + 2$ and $f(x) = g(h(x))$. 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.

Lab Requirements

The AP Computer Science A course must include a minimum of 20 hours of hands-on structured lab experiences to engage students in individual or group problem solving. Thus, each AP Computer Science A course includes a substantial laboratory component in which students design solutions to problems, express their solutions precisely (e.g., in the Java programming language), test their solutions, identify and correct errors (when mistakes occur), and compare possible solutions.

Computer Language

The AP Computer Science A course requires that solutions of problems be written in the Java programming language. Because the Java programming language is extensive with far more features than could be covered in a single introductory course, the AP Computer Science A Exam covers a subset of Java. The AP Java subset can be found in Appendix A of the Course Description.

Goals of AP Computer Science A

Students should be able to

- Design, implement, and analyze solutions to problems;
- Use and implement commonly used algorithms;
- Develop and select appropriate algorithms and data structures to solve new problems;
- Write solutions fluently in an object-oriented paradigm;
- Write, run, test, and debug solutions in the Java programming language, utilizing standard Java library classes and interfaces from the AP Java subset;
- Read and understand programs consisting of several classes and interacting objects;
- Read and understand a description of the design and development process leading to such a program; and
- Understand the ethical and social implications of computer use.

Topic Outline for AP Computer Science A

- I. Object-Oriented Program Design
 - A. Program and class design
- II. Program Implementation
 - A. Implementation techniques
 - B. Programming constructs
 - C. Java library classes and interfaces included in the AP Java Subset
- III. Program Analysis
 - A. Testing
 - B. Debugging
 - C. Runtime exceptions
 - D. Program correctness
 - E. Algorithm analysis
 - F. Numerical representations of integers
- IV. Standard Data Structures
 - A. Primitive data types (int, boolean, double)
 - B. Strings
 - C. Classes
 - D. Lists
 - E. Arrays (1-dimensional and 2-dimensional)
- V. Standard Operations and Algorithms
 - A. Operations on data structures
 - B. Searching
 - C. Sorting
- VI. Computing in Context
 - A. System reliability
 - B. Privacy
 - C. Legal issues and intellectual property
 - D. Social and ethical ramifications of computer use

AP Computer Science A Exam Structure

AP COMPUTER SCIENCE A EXAM: 3 HOURS

Assessment Overview

All code on the AP Computer Science A Exam is consistent with the AP Java subset that can be found in Appendix A of the Course Description. All questions involving code should be answered in Java. Students are not tested on minor points of syntax.

Format of Assessment

Section I: Multiple Choice | 40 Questions | 1 Hour, 30 Minutes |
50% of Exam Score

- Discrete Question topics will include: programming fundamentals, data structures, logic, algorithms/problem solving, object-oriented programming, recursion, and software engineering.

Section II: Free Response | 4 Questions | 1 Hour, 30 Minutes |
50% of Exam Score

- Short Answer (each requiring Java programming language)
- Solve problems involving more extended reasoning.

AP COMPUTER SCIENCE A SAMPLE EXAM QUESTIONS

Sample Multiple-Choice Question

A car dealership needs a program to store information about the cars for sale. For each car, they want to keep track of the following information: number of doors (2 or 4), whether the car has air-conditioning, and its average number of miles per gallon. Which of the following is the best object-oriented program design?

- (a) Use one class, Car, with three instance variables: int numDoors, boolean hasAir, and milesPerGallon.
- (b) Use four unrelated classes: Car, Doors, AirConditioning, and MilesPerGallon.
- (c) Use a class Car with three subclasses: Doors, AirConditioning, and MilesPerGallon.
- (d) Use a class Car, with a subclass Doors, with a subclass AirConditioning, with a subclass MilesPerGallon.
- (e) Use three classes: Doors, AirConditioning, and MilesPerGallon, each with a subclass Car.

Correct Answer: A

Sample Free-Response Question: Experimental Design

A travel agency maintains a list of information about airline flights. Flight information includes a departure time and an arrival time. You may assume that the two times occur on the same day. These times are represented by objects of the Time class.

The declaration for the Time class is shown below. It includes a method minutesUntil, which returns the difference (in minutes) between the currentTime object and anotherTime object.

Students are provided with a scenario and part of the program to solve that scenario. They are then asked to complete a method in that program so that the program can calculate the duration of flights.