AP Computer Principles
AND
AP Computer Science A

Overview of Curriculum Framework

AP

1. Creativity

2. Abstraction

3. Data and Information

4. Algorithms

5. Programming

6. The Internet

7. Global Impact

CollegeBoard

15

Pacing Overview

Exploration Theme	Hours of Exploration	Exploration Theme Summary
0: Introduction to CSP	15	Computer science is introduced as a study of the seven big ideas facilitated by the application of the six computational thinking practices. Websites inspire discussions and document creation related to current topics. Students read and discuss articles, respond in journal entries, and create artifacts that summarize their interests and knowledge.
1: Creativity and Computing	25	Students write program code, primarily in JavaScript, throughout the school year. Much of this programming focuses on creating artifacts of interest to individual students. Manipulations range from modifying all pixels in an image to performing algorithmic computations associated with changes in selected collections of pixels. Students write programs to merge images using pixel selection algorithms, create patterns, and combine multiple techniques into a new, higher-level technique. Students analyze and visualize data sets, explore JavaScript application programming interfaces, and create a dynamic website that facilitates the exploration of topics of personal interest.
2: Identifying and Using Abstractions	10	Students are introduced to abstractions used to efficiently create effective programs. Everything that a student says or does involves the use of abstractions. It is important to help students become aware of abstractions and how they affect their ability to understand and navigate the world in which they live. In particular, Computer Science Principles involves abstractions whose application enhances computational capabilities.
3: Using Data	10	Programming is used to help students effectively process and summarize data. Publicly available data sources are used for exploration, the discovery of information, and the creation of new knowledge. Students identify topics that interest them and search the Web for data associated with those topics. Early in the course, students are provided data sources that are stored on the server used for their programming. This method allows students to more easily process data in an effort to discover new information and knowledge. Later in the course, data sources must be entered into the programming environment in an acceptable format. Doing so often requires computational manipulation of the data using text editors, spreadsheets, JavaScript programs, or statistics software packages.

Exploration Theme	Hours of Exploration	Exploration Theme Summary
4: Exploring Algorithms	10	Students present, analyze, and implement algorithms that are designed to accomplish specific tasks related to solving problems of personal interest and motivational artifact creation. While exploring image manipulations, students learn how to create gray scale images, produce enhanced color schemes based on original image color schemes, merge images, and transform images using reflections, rotations, and dilations. Students also generate row, column, diagonal, and checkerboard patterns. They use their knowledge and experience to create new algorithms that accomplish their image manipulation goals. Students also apply their knowledge of algorithms to process data and gain new information and knowledge.
5: Problem Solving With Programming	20	Students use programming to efficiently solve problems. Relatively simple problems that require programming to efficiently obtain a solution are presented to students early in the course. For example, students are asked questions about images in terms of their color properties. Because the images contain thousands of pixels, the students must use computational thinking to answer the questions in a reasonable amount of time. Students are required to write programs that efficiently answer the questions, and they are challenged to pose questions that they and their peers can further explore.
6: Guided Internet Explorations	15	Students explore the characteristics and uses of the Internet. They examine and discuss security and privacy issues related to the Internet. For example, students simulate a binary transmission using paper clips. They create and use Huffman coding trees to encode text messages into binary form and then model the messages using chains or packets of connected paper clips. Students also explore encryption and decryption techniques.
7: Identifying Innovations	20	Students select areas of interest that are affected by computing. They research and report on those areas in written and oral reports and present computational artifacts. Past activities have included a debate about the impact of computing on society, the creation of videos on computing in a variety of fields, and the development of infographics that illustrate computing information that is personally relevant.

Overview of Assessments

Through-Course Assessment (Weight = 40%)

Performance Task	Individual Weight
Explore: Implications of Computing Innovations	16%
Create:	24%

End-of-Course AP® Exam (Weight = 60%)

Item Type	Number	Timing
Multiple Choice (Single- and multiple-select)	74 questions	120 minutes

Summer Work 2018 – AP Computer Science Principles

This is an introductory course intended to provide an overview of the concepts used in computer science.

This course uses JavaScript programming. It is requested that student work through and complete the Khan Academy section "Computer Programming". (https://www.khanacademy.org/computing/computer-programming)

They should create an account and track their progress. Paying close attention to the JavaScript programming concepts.

About the Course

AP Computer Science Principles offers a multidisciplinary approach, focusing on the creative aspects of programming.

Students will learn:

▸ Creative problem solving

▸ How to apply computational processes to analyze large data sets

▸ Programming

▸ How the Internet works and important cyber security issues

▸ Impacts of computing innovations

"Every student has this ability to create things that they didn't even imagine they could create. And when they do, it starts to draw them in and inspire them to think well beyond what they even imagined."

—**Richard Kick**

AP® Computer Science Principles Teacher

◇ CollegeBoard

Two AP® Computer Science Courses

A^P

	Computer Science A	Computer Science Principles
Curricular Focus	Problem solving and object-oriented programming	Big ideas of computer science (including programming)
Programming Language	Java	Teachers choose
Course Goal	Encourage skill development among students considering future studies & career in computer science or other STEM fields	Encourage broader participation in CS & STEM, including AP CSA
Assessment Experience	▶ Multiple-choice and free-response questions	▶ Multiple-choice exam ▶ Two performance tasks administered by the teacher, and students submit digital artifacts
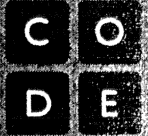
CS Principles Course Snapshot

Unit 1 - The Internet

Ch. 1: Digital Information

wk	
1	Personal Innovations Sending Binary Messages Sending Messages with the Simulator
2	Number Systems Binary Numbers Sending Numbers
3	Encoding and Sending Formatted Text Unit 1 Chapter 1 Assessment

Ch. 2: Inventing The Internet

wk	
	The Internet is for Everyone
4	The Need for Addressing Routers and Redundancy Packets and Making a Reliable Internet
5	The Need for DNS HTTP and Abstraction Practice PT - The Internet and Society Unit 1 Chapter 2 Assessment

Unit 2 - Digital Information

Ch. 1: Encoding and Compressing Info

wk	
1	Bytes and File Sizes Text Compression Encoding B&W Images
2	Encoding Color Images Lossy Compression and File Formats
3	Encode an Experience Unit 2 Chapter 1 Assessment

Ch. 2 - Manipulating and Visualizing Data

wk	
4	Intro to Data Finding Trends with Visualizations Check Your Assumptions Good and Bad Data Visualizations
5	Making Data Visualizations Discover a Data Story
6	Cleaning Data Creating Summary Tables Practice PT - Tell a Data Story Unit 2 Chapter 2 Assessment

Unit 3 - Intro to Programming

Ch. 1 - Programming Languages & Algorithms

wk	
1	The Need For Programming Languages The Need for Algorithms Creativity in Algorithms
2	Using Simple Commands Creating Functions Functions and Top-Down Design

Unit 3 - Intro to Programming (con'd)

wk	
3	APIs and Function Parameters Creating functions with Parameters Looping and Random Numbers
4	Practice PT - Design a Digital Scene Unit 3 Chapter 1 Assessment

Unit 4 - Big Data and Privacy

Ch. 1: The World of Big Data and Encryption

wk	
1	What is Big Data? Rapid Research - Data Innovations Identifying People with Data
2	The Cost of Free Simple Encryption
3	Encryption with Keys and Passwords Public Key Crypto Rapid Research - Cybercrime
4	Practice PT - Big Data and Security Dilemmas Unit 4 Chapter 1 Assessment

Unit 5 - Building Apps

Ch. 1: Event-Driven Programming

wk	
1	Buttons and Events Multi-screen Apps Building an App: Multi-Screen App
2	Controlling Memory with Variables Building an App: Clicker Game Unit 5 Assessment 1 User Input and Strings
3	"If" Statements Unplugged Boolean Expressions and "If" Statements
4	"If-else-if" and Conditional Logic Building an App: Color Sleuth Unit 5 Assessment 2

Ch. 2: Programming with Data Structures

	While Loops Loops and Simulations Introduction to Arrays
	Building an App: Image Scroller Unit 5 Assessment 3 Processing Arrays Functions with Return Values
	Building an App: Canvas Painter Unit 5 Assessment 4 Practice PT - Create Unit 5 Assessment 5 - AP Pseudocode Practice

Performance Tasks

1 hr	Tech Setup (Can be completed at any time)
10 hrs	Explore prep (Can be completed after Unit 4) Explore PT (8 class hours)
14 hrs	Create Prep(Can be completed after Unit 5 Chapter 1) Create PT (12 class hours)