AP Computer Science Principles

Mr. Browne
SBrowne@efsd.net
412.896.2300 Ext 7911

Introduction

AP Computer Science Principles is the newest AP® course from the College Board. This course introduces students to the foundational concepts of computer science and explores the impact computing and technology have on our society.

With a unique focus on creative problem solving and real-world applications, the CodeHS AP Computer Science Principles course gives students the opportunity to explore several important topics of computing using their own ideas and creativity, use the power of computing to create artifacts of personal value, and develop an interest in computer science that will foster further endeavors in the field.

Course Overview

Prerequisites: There are no official prerequisites for the CodeHS AP Computer Science Principles course. This course is meant to be a first-time introduction to computer science and does not require students to come in with any computer programming experience. However, we recommend that students take our Introduction to Computer Science prior to our AP courses (more info at codehs.com/library). Students who have completed our Intro to CS course will be able to apply knowledge of concepts covered in the Intro course to the more advanced setting of the AP courses. We also recommend that students complete a first-year high school algebra course prior to taking this course. Students should be comfortable with functions and function notation such as $f(x) = x + 2$ as well as using a Cartesian (x, y) coordinate system to represent points in a plane.

Learning Environment: The course utilizes a blended classroom approach. The content is a mix of web-based and physical activities. Students will write and run code in the browser, create websites and digital artifacts, and engage in in-person collaborative exercises with classmates. Teachers utilize tools and resources provided by CodeHS to leverage time in the classroom and give focused 1-on-1 attention to students. Each unit of the course is broken

down into lessons. Lessons consist of video tutorials, short quizzes, example programs to explore, written programming exercises, free response exercises, collaborative creation projects, and research projects.

Programming Environment: Students write and run programs in the browser using the CodeHS editor. Students will be able to write both text-based and block-based JavaScript programs, and students will use Processing.js to create graphical programs. They will also create webpages using HTML, CSS, and JavaScript. These webpages will be hosted on the CodeHS website so that they can keep a running portfolio of their creative projects, and easily share their programs with the world. Students gain programming experience early on in the course that will enable them to explore the rest of the course topics through computational thinking practices.

Quizzes: At the end of most units, students take a summative multiple choice unit quiz in the style of the AP Exam that assesses their knowledge of the concepts covered in the unit. The course also provides an AP Test Practice unit with a cumulative AP Practice Multiple Choice Test.

Course Objectives

This course is based directly on the College Board AP Computer Science Principles Framework. We recommend reading the curriculum framework here for context. The main course objectives are summarized below in the six computational thinking practices and seven big ideas for the course.

Computational Thinking Practices:
The six computational thinking practices represent important aspects of the work that computer scientists engage in, and are denoted here by P1 through P6:

- P1: Connecting Computing
    - Identify impacts of computing.
    - Describe connections between people and computing.
    - Explain connections between computing concepts.
- P2: Creating Computational Artifacts
    - Create an artifact with a practical, personal, or societal intent.
    - Select appropriate techniques to develop a computational artifact.
    - Use appropriate algorithmic and information management principles.
- P3: Abstracting
    - Explain how data, information, or knowledge is represented for computational use.
    - Explain how abstractions are used in computation or modeling.
    - Identify abstractions.
    - Describe modeling in a computational context.

- P4: Analyzing Problems and Artifacts
    - Evaluate a proposed solution to a problem.
    - Locate and correct errors.
    - Explain how an artifact functions.
    - Justify appropriateness and correctness of a solution, model, or artifact.
- P5: Communicating
    - Explain the meaning of a result in context.
    - Describe computation with accurate and precise language, notations, or visualizations.
    - Summarize the purpose of a computational artifact.
- P6: Collaborating
    - Collaborate with another student in solving a computational problem.
    - Collaborate with another student in producing an artifact.
    - Share the workload by providing individual contributions to an overall collaborative effort.
    - Foster a constructive, collaborative climate by resolving conflicts and facilitating the contributions of a partner or team member.
    - Exchange knowledge and feedback with a partner or team member.
    - Review and revise their work as needed to create a high-quality artifact.

Big Ideas:

The seven big ideas of the course encompass foundational ideas in the field of computer science, and are denoted here by B1 through B7:

- B1: Creativity
    - How can a creative development process affect the creation of computational artifacts?
    - How can computing and the use of computational tools foster creative expression?
    - How can computing extend traditional forms of human expression and experience?
- B2: Abstraction
    - How are vastly different kinds of data, physical phenomena, and mathematical concepts represented on a computer?
    - How does abstraction help us in writing programs, creating computational artifacts, and solving problems?
    - How can computational models and simulations help generate new understanding and knowledge?
- B3: Data and Information
    - How can computation be employed to help people process data and information to gain insight and knowledge?
    - How can computation be employed to facilitate exploration and discovery when working with data?

    - What considerations and tradeoffs arise in the computational manipulation of data?
    - What opportunities do large data sets provide for solving problems and creating knowledge?
- B4: Algorithms
    - How are algorithms implemented and executed on computers and computational devices?
    - Why are some languages better than others when used to implement algorithms?
    - What kinds of problems are easy, what kinds are difficult, and what kinds are
    - impossible to solve algorithmically?
    - How are algorithms evaluated?
- B5: Programming
    - How are programs developed to help people, organizations, or society solve problems?
    - How are programs used for creative expression, to satisfy personal curiosity, or to create new knowledge?
    - How do computer programs implement algorithms?
    - How does abstraction make the development of computer programs possible?
    - How do people develop and test computer programs?
    - Which mathematical and logical concepts are fundamental to computer programming?
- B6: The Internet
    - What is the Internet? How is it built? How does it function?
    - What aspects of the Internet's design and development have helped it scale and flourish?
    - How is cybersecurity impacting the ever-increasing number of Internet users?
- B7: Global Impact
    - How does computing enhance human communication, interaction, and cognition?
    - How does computing enable innovation?
    - What are some potential beneficial and harmful effects of computing?

The AP Performance Tasks:
The through course assessment is a set of performance tasks designed to gather evidence of student proficiency in the learning objectives. The AP Performance Tasks (PTs) are in-class assessments, administered by the teacher, that allow students to exemplify their learning through authentic, "real-world" creations. For more information about the AP Performance Tasks, refer to the curriculum framework.

The two performance tasks as defined by College Board are:

1. Explore - Implications of Computing Inventions
    - Students explore the impacts of computing on social, economic, and cultural areas of our lives.
2. Create - Applications from Ideas
    - Students create computational artifacts through the design and development of programs.

Students will gain the experience necessary to complete the PTs in class. Each unit comes with practice PTs in which students will research topics in computing, and create their own digital artifacts. Students will create and maintain a website that will hold each student creation throughout the course. This will serve as a running portfolio of each creative project the student completes. Sufficient time is set aside in the course for students to prepare for and complete both PTs.

The AP Exam:
This course will prepare students for the multiple-choice AP Computer Science Principles examination. Each lesson comes with quizzes to test essential knowledge for the AP Exam. Each unit includes a cumulative AP style multiple-choice exam to test understanding of the concepts in the unit and provide immediate feedback to the student.

Start of Class

Upon arrival, there will be a bell ringer projected on the screen that you will complete on Canvass with your iPad. You will have 3 minutes to complete this and submit it. This will be graded every day. In-class attentiveness and participation is vital to be successful in this class. You must remain engaged and not interfere with the learning of anyone else in the class. Class periods will end with an exit ticket.

Electronics Policy

iPads are to be used during class only for educational purposes when permitted to do so. Under no circumstances should students be checking messages, playing games or texting during class. Keep cell phones in your pocket/purse. I do not want to see them in class.

Grading Policy: Late assignments, Absences and Make-up Work

Bell ringers are graded daily and will be 10% of your grade. Homework and in-class work will comprise 30% of your grade. Labs and projects will account for another 40% of your grade. Quizzes and exams will make up the last 20%. Late assignments will be accepted up to five school days past the due date, after which a zero will be recorded in the grade book. A 50% penalty will be assessed on all late assignments. If absent, students have as many days to make up a quiz or test as the number of school days they missed. It is the students' responsibility to check with me when they return from an absence to check for missed work, notes, quizzes and tests. A zero will be given for any assignment if the student's absence is unexcused.

Extra Help

AP CSP can be challenging, but the problem solving skills and real-world skills you learn in this classroom will benefit you no matter what career you end up in. To make sure that you are successful in this course, I will be in my classroom every Tuesday and Thursday from 2:30 to 3:00 for any student that wants help. Please take advantage of this and do not wait until you fall too far behind. I am more than willing to provide any extra help that you may need. If another time fits in better with your schedule, please let me know and we can arrange an alternative time. Stay current with the homework and ask for help whenever you encounter difficulties. I want you all to succeed.

Remind

I will be using a free app called Remind to send out general class announcements as well as homework and test reminders. There is an app you can download as well. I will give instructions and the class code during class.

AP Computer Science Principles

Academic Integrity

These standards have been adopted by the science department and will apply to all science classes.

- Try! Listen! Think!
- Be prepared for your class
- Use your iPad and phone for educational purposes only
- Do not get off task or give up
- Do not discuss test questions or answers
- Follow directions
- Be safe in the lab
- Do not cheat
- Do not damage school property

***Please keep the first six pages in your notebook and return only this signature sheet.**

I have read and understand the above information

Student name: _____

Student signature: _____

Parent or guardian name: _____

Parent or guardian signature: _____

Parents,

I will be contacting you at various times throughout the year with progress reports and updates regarding your child's standing in this class. Therefore, I need to know your email address and/or telephone number(s). Please also indicate the best time to call if I need to speak with you directly.

Email: _____

Telephone #: 1.) _____

2.) _____

Best time to call: _____