Manual Arts High School - College Preparatory Magnet
Los Angeles Unified School District

AP Computer Science Principles Syllabus (2020-21)

Instructor: José Aguilar
Pronouns: He/Him/His
Email: jose.aguilar3@lausd.net
Office Hours: TTH 2:15 - 4:00 PM or by appointment

Welcome to AP CSP!

Course Description:
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. The course is designed with the goal of creating leaders in computer science fields and attracting and engaging those who are traditionally underrepresented with essential computing tools and multidisciplinary opportunities.

Manual Arts College Preparatory Magnet Mission:
The mission of the College Preparatory Magnet program at Manual Arts High School is to prepare students to change their world positively. Magnet students will possess the knowledge and skills necessary to pursue their academic and career goals, to compete successfully in the world market, and to be critical, analytical, life-long learners.

Manual Arts College Preparatory Magnet Vision:
Students will be active, engaged, and informed citizens to meet the challenges of a global society.

Mr. Aguilar's Expectations:
Its okay to be wrong!
PLEASE ASK QUESTIONS!

Class Policies:

1. Absenteeism: You are responsible for making up missing work. You will be responsible for making up any assignments and watching any session videos missed. Live videos sessions will NOT be recorded.

2. Homework: Live instruction will focus on content; students will be given assignments that may include videos, digital textbook, College Board assessments, etc. You are responsible for completing all readings, assessments and studying all content covered in class.
3. Academic Support/Tutoring: Tuesday's will be academic support; it is the opportunity for you to ask questions and get help. As well as office hours.
4. Contact: You are responsible for staying in contact with the teacher. Please reach out if you need assistance with anything, using the communication protocol.

Materials:

- Interactive Journal
- Notebook(Physical/Digital)
- Computer w/ Internet Access

Texts:

- Blown to Bits: Your Life, Liberty, and Happiness After the Digital Explosion (Abelson, Ledeen, Lewis)
- Race After Technology: Abolitionist Tools for the New Jim Code (Ruha Benjamin)
- Stuck in the shallow end: Education, Race and Computing (Jane Margolis)

Curriculum:

Code.org - Starting in the 2016, Code.org is recognized by the College Board as an endorsed provider of curriculum and professional development for AP® Computer Science Principles (AP CSP). This endorsement affirms that all components of Code.org CSP's offerings are aligned to the AP Curriculum Framework standards and the AP CSP assessment. Using an endorsed provider affords schools access to resources including an AP CSP syllabus pre-approved by the College Board's AP Course Audit, and officially recognized professional development that prepares teachers to teach AP CSP.

Khan Academy - Khan Academy is recognized by the College Board as an endorsed provider of curriculum for AP® Computer Science Principles (AP CSP). This endorsement affirms that all components of Khan Academy CSP's offerings are aligned to the AP Curriculum Framework standards and the AP CSP assessment. Using an endorsed provider affords schools access to resources including an AP CSP syllabus pre-approved by the College Board's AP Course Audit, and officially recognized professional development that prepares teachers to teach AP CSP.

Methodology:

This course will combine different methods of instruction, including class lecture (show and tell), demonstration, individual practice, small groups activity, one on one conference with the instructor, homework, class presentations, related media demonstrations, use word definition, word wall, thinking maps, inquiry based lessons, direct instruction, gallery walk. etc.

Methods of Assessment:

- Worksheets and Stages on CODE.ORG & Khan Academy
- Tests and quizzes

- Classroom discussion and Participation
- Programming Assignment & Projects (Performance Task Practices)
- Group Work
- Presentations (One on one, PAIR and SHARE, group or whole class)

Course Outline:

Unit	Description
Unit 1 Digital Information	Students explore the way computers store and represent complex information like numbers, text, images, and sound. The unit begins with students investigating what it means to represent information, and challenges students to design their own representation systems. Students then learn the ideas behind real-world systems used to represent complex information. Later lessons focus on the challenges that arise from digitizing information, such as the need to compress it, or questions of intellectual property. The unit project emphasizes the profound impact digital information has on modern life.
Unit 2 The Internet	Students learn how the Internet works and discuss its impacts on politics, culture, and the economy. This unit heavily features the Internet Simulator, a tool designed to let students see, use, and explore the way different layers of the internet work. Through a series of activities that build on one another, students investigate the problems the original designers of the internet had to solve and then "invent" their own solutions. At the conclusion of the unit, students research an "Internet Dilemma," both from the standpoint of its technical background and its impacts on different groups of people.
Unit 3 Intro to App Design	Students design their first app while learning both fundamental programming concepts and collaborative software development processes. Students work with partners to develop a simple app that teaches classmates about a topic of personal interest. Throughout the unit, they learn how to use Code.org's programming environment, App Lab, to design user interfaces and write simple event-driven programs. Along the way, students learn practices like debugging, pair programming, and collecting and responding to feedback, which they will be able to use throughout the course as they build ever more complex projects. The unit concludes with students sharing the apps they develop with their classmates.
Unit 4 Variables, Conditionals, and Functions	Students expand the types of apps they can create as they learn how to store information (variables), make decisions (conditionals), and better organize code (functions). Each programming topic is covered in a specific sequence of lessons that ask students to 'Explore' ideas through hands-on activities, 'Investigate' these ideas through guided code reading, 'Practice' with sample problems, and apply their understanding as they 'Make' a one-day scoped project. The entire unit concludes with a three-day open-ended project in which students must build an app that makes a recommendation about any topic they wish.
Unit 5 Lists, Loops, and Traversals	Students learn to build apps that use and process lists of information. Like the previous unit, students learn the core concepts of lists, loops, and traversals through a series of EIPM lesson sequences. Later in the unit, students are introduced to tools that allow them to import tables of real-world data to help further power the types of apps they can make. At the conclusion of the unit, students complete a week-long project in which they must design an app around a goal of their choosing that uses one of these data sets.
Unit 6 Algorithms	Students learn to design and analyze algorithms to understand how they work and why some algorithms are considered more efficient than others. This short unit is entirely unplugged, and features hands-on activities that help students get an intuitive sense of how quickly different algorithms run and the pros and cons of different algorithms. Later in the unit, students explore

	concepts like undecidable problems and parallel and distributed computing.
Unit 7 Parameters, Return, and Libraries	Students learn how to design clean and reusable code that can be shared with a single classmate or the entire world. In the beginning of the unit, students are introduced to the concepts of parameters and return, which allow for students to design functions that implement an algorithm. In the second half of the unit, students learn how to design libraries of functions that can be packaged up and shared with others. The unit concludes with students designing their own small library of functions that can be used by a classmate.
Unit 8 Create PT Prep	This short unit prepares students to complete the AP® Create Performance Task (PT). Students will have learned the skills and concepts necessary to complete the task in previous units and will even have seen components of the task itself. This unit fully explains all components of the task and walks students through completing and submitting it.
Unit 9 Data	Students explore and visualize datasets from a wide variety of topics as they hunt for patterns and try to learn more about the world around them from the data. Once again, students work with datasets in App Lab, but are now asked to make use of a data visualizer tool that assists students in finding data patterns. They learn how different types of visualizations can be used to better understand the patterns contained in datasets and how to use visualizations when investigating hypotheses. At the conclusion of the unit, students learn about the impacts of data analysis on the world around them and complete a final project in which they must uncover and present a data investigation they've completed independently.
Unit 10 Cybersecurity and Global Impacts	Students research and debate current events at the intersection of data, public policy, law, ethics, and societal impact in the final unit of the course. This unit is built around a simulated "future school" conference in which students must take on the persona of a stakeholder in a school setting and propose and debate technological innovations that could improve schools. Throughout the unit, students learn about the privacy and security risks of many computing innovations, and learn about the ways some of these risks can be mitigated. Students complete their Explore Curricular Requirement as part of this project as they investigate at least three computing innovations, then discuss and debate many others with their classmates. At the conclusion of the unit, the class holds a conference in which teams present their overall vision for a school of the future and the computing innovations that would power it.

Grading:

Breakdown of Assessment -

Performance Tasks & Programming Assignments - 30%
End of Unit Assessments - 20%
Presentation & Essays - 20%
Worksheets/Stages - 15%
Participation/Discussions - 15%

Scale - Scoring Guide*:

A = 90-100 %
B = 80-89 %
C = 70-79 %
D = 60-69 %
F = 0-59 %

*Based on Class Final Semester Performance, curves may be applied to hurt not help you.

Late Work - Communication is key, if you need an extension or overall help with assignment reach out. Communicate before its due.