Dominican International School
AP Computer Science Principles
SY: 2022-23

Grade Level 11/12   Teacher Tim Sampson
1 Year   Email: tsampson@dishs.tp.edu.tw

Description

As described by the College Board, AP Computer Science Principles offers a multidisciplinary approach to teaching the underlying principles of computation. The course will introduce students to the creative aspects of programming, abstractions, algorithms, large data sets, the Internet, cybersecurity concerns, and computing impacts. AP Computer Science Principles also gives students the opportunity to use current technologies to create computational artifacts for both self-expression and problem solving. Together, these aspects of the course make up a rigorous and rich curriculum that aims to broaden participation in computer science.

CS50 is Harvard University's introduction to the intellectual enterprises of computer science and the art of programming for students with a diversity of technological background and experience. CS50 for AP Computer Science Principles is an adaptation of CS50 specifically tailored to align with the AP Computer Science Principles curriculum framework. The course assignments, materials, and resources are all identical to the version of the course taught at the college-level, albeit adapted to suit a secondary school audience.

This Syllabus has been adapted from the CS50 APCS pre approved syllabus CS50 submitted for approval to the College Board. Our course will follow the original approved syllabus, with supplemental content and exercises from Code.org AP CSP curriculum MakeCode AP CSP and Khan Academy AP CSP.

The Big Ideas and Thinking practices

Seven Big Ideas

Big Idea 1: Creativity
Big Idea 2: Abstraction
Big Idea 3: Data
Big Idea 4: Algorithms
Big Idea 5: Programming
Big Idea 6: The Internet
Big Idea 7: Global Impacts

Six Computational Thinking Practices

P1: Connecting Computing
P2: Creating Computational Artifacts
P3: Abstracting
P4: Analyzing Problems and Artifacts
P5: Communicating
P6: Collaborating

Requirements

It is recommended that a student in the AP Computer Science Principles course should have successfully completed a first year high school algebra course with a strong foundation on basic linear functions and

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 in a plane. It is important that students and their advisers understand that any significant computer science course builds upon a foundation of mathematical and computational reasoning that will be applied throughout the study of the course. AP Computer Science Principles Course Overview

How this course can benefit students

Among this course's objectives is to supply students with a comprehensive introduction to the fundamentals of the discipline of computer science. We will do so using programming in several different languages as a vehicle to introduce these fundamentals, including such topics as algorithms, abstraction, data, global impact, and internet technologies. Though the course is programming-heavy, it should be stressed that this is not a "programming course"; rather, this course should be considered one of problem-solving, creativity, and exploration. By year's end, students will have a richer understanding of the key principles of the discipline of computer science. They will be able to speak intelligently about how computers work and how they enable us to become better problem-solvers, and will hopefully be able to communicate that knowledge to others.

Whether students elect to take no other computer science courses in their lifetime or consider this class the first step in a longer course of study, it is our sincere hope that students feel more comfortable with—and indeed sometimes skeptical of—the technologies that surround us each day. CS50 APCS pre approved syllabus

Primary Textbook and Other Resources

Books and Online Instructional Material

- CS50 Reference
- Kahn AP® Computer Science Principles Review
- CS Principles: Big Ideas in Programming

Videos

- Code.org Video library Topical Videos
- This is CS50 Youtube.com Lectures, seminars, shorts and walkthroughs
- PBS Crash Course Computer Science

Programming Environment
- Visual Studio Code for CS50

Assessment

Assignments 30%
Quizzes 30%
Quarter Exam 30%
Deportment 10 %

Additional Information:

Course Design and Labs

This course is a four quarter course, with approximately 34 weeks of instructional time before the AP exams. All classes will regularly take place in the school's computer lab, except on occasions when students will be completing on off-line activities. Students will meet for five 45 minute class periods per week, which are arranged in two class blocks 3 times per week. A topic is usually completed within two weeks. The time after the AP exam will be used for enrichment activities and project work. In addition to the graded activities and labs there will be a test after each unit and also a final exam each quarter that covers all material covered up to that point in time.

Typically, class starts with a short introductory lecture and is followed by time allotted for students to complete practice questions and programming challenges, Usually individually, but sometimes in pairs or small groups.

This syllabus has been adapted from the AP50 Syllabus to accommodate our student's CS background and school schedule. APCSP CS50 syllabus can be accessed https://cs50.harvard.edu/ap/2023/syllabus/

AP CS A Exam

Students who complete this course will be prepared to take the AP CS A Exam in May.

LTO's D'TORCH (Truthful, Organized, Reflective, Courageous and Helpful)

In CS classes the categories of the D'TORCH most practiced and assessed are:
- Organized - Students utilize Google Classroom to edit, submit and keep track of their assignments.
- Reflective - Students will regularly write activity reflections in their online journal.
- Helpful - Students are empowered to ask for and provide explanations and give examples to help classmates through particularly difficult problems.

Class Expectations
- Come to class on time and be prepared
- Have a positive attitude and be willing to learn.
- Respect yourself, others, and our school.
- Always complete your work and try your best.
- Actively participate, listen carefully, but don't speak out of turn.
- All assignments must be completed.

Homework and Quiz Rules
- All assignments must be turned in on the day they are due.
- 1 day late = Maximum of only 60%
- 2+ days late = Project-I & Only 60%
- If a student has been absent, it is his/her duty to find out what work is due, and hand it in a day later.
- All assignments must satisfactorily be completed.
- If you are absent on the day of a quiz, you will only be able to get a maximum of 60%.

Classroom Rules
- All students are expected to follow the rules. Consequences will follow if rules are broken.
- Read and follow the standard school rules.
- Be on time and neatly dressed, in full school uniform.
- Speak in ENGLISH ONLY.
- Respect your teachers, fellow students and their property.
- Keep your seating space and classroom clean and neat.
- No eating or drinking in the ICT Labs.
- Ask permission to leave the class.

- Neither cheating nor copying in any form will be accepted. Anyone caught doing either during an activity, project or assessment will be given a zero.

Academic Honesty (Plagiarism)

HS CS at DIS is adopting the Harvard CS guidelines on academic honesty. From the Harvard Syllabus

"The essence of all work that you submit to this course must be your own. Collaboration on problems is not permitted (unless explicitly stated otherwise) except to the extent that you may ask classmates and others for help so long as that help does not reduce to another doing your work for you. Generally speaking, when asking for help, you may show your code or writing to others, but you may not view theirs, so long as you and they respect this policy's other constraints."

More details can be found at this link: https://cs50.harvard.edu/ap/2023/syllabus/#academic-honesty

WEEKLY SCHEDULE

(NB: Depending on time and interest, the teacher may delete and/or add other selections.)	
Week / Date	Topic / Projects / Assessments
Week 1 **Aug 10th to 12th** **3 Days of Class** *10~ First Day / Orientation Day*	**Module: Hardware** **Lecture: Hardware** In this module, students learn about how it all works underneath the hood and how to solve problems when something goes wrong. **Welcome Unit 1 Day 01 Intro** **Lecture: Hardware** **Around the House:** In this writing assignment you will attempt to find and describe something in your house that on first inspection might not be a computer, but on closer inspection it meets the criteria and is indeed a computer. Be sure to not choose anything obvious, such as a mobile phone or a smart tv.
Week 2 **Aug 15th to 19th** *Opening Mass*	**Module: Hardware, continued** CS50's Understanding Technology **Tech Spotlight:** In this writing problem, we're going to try to home in on some of the corners of the Internet to keep your eye on as we begin to explore computer science in more depth.Pick some piece of technology with which you are familiar and that you might use everyday. **Everyday Algorithms:** Write up algorithms to exhaustively describe in text and pseudocode, steps to brush one's teeth, eat an orange, and something you do everyday. **Build a PC Virtually:** First we take a look at what sort of hardware we are currently using to do our school work or play games and then we go to a website where we can set a budget and virtually build a pc using modern specifications.
Week 3 **Aug 22nd to 26th**	**Module: Programming with Scratch** **Lecture: Programming with Scratch**

	In this module, students will learn about how data is represented in their computer and the language of computers – binary, how information is encoded so that humans can understand it and begin to explore the ways in which computers process information.
	Create with Scratch: Students will use the drag-and-drop programming language called Scratch to implement a project of their choice (be it an animation, a game, interactive art, or anything else)
Week 4 **Aug 29th to Sep 2nd**	**Module: Programming with C** **Lecture: Programming with C** In this module, students will learn the fundamentals of computer programming, to permit them to begin to manipulate information and data and command a computer to do calculations they wish for it to perform. **PSET Hello:** Students learn the syntax specific to C. Here, they write their first program in a web-based programming environment called the CS50 IDE. **PSET Fahrenheit:** Students will write a program that converts a temperature in Celsius to Fahrenheit and explore bugs that might arise when dealing with imprecision relating to floats and division in C. **Lab Scrabbl**e: Students will write a program which determines which of two Scrabble words is worth more.
Week 5 **Sep 5th to 9th** **4 Days of Class** *8~ Mass &Birthday Mother Mary* *9th – Moon Festival*	**Module: Programming with C, continued** **PSET Cash:** This activity introduces students to greedy algorithms. Here, they will write a program that first asks the user how much change is owed and then outputs the minimum number of coins with which said change can be made. **PSET Pennies:** Students will create a file that calculates the amount that the user will have received in total by the end of the month (not just on the last day) if some initial amount is doubled on every day but the first, expressed not as pennies but as dollars and cents. **PSET ISBN:** Students will further build on the abstractions available to them in C. Here, they will explore iteration and loops by writing a program that prompts the user for an ISBN-10 and then reports (via printf) whether the number's legitimate. The program's last line of output should be either yes or no, nothing more, nothing less.
Week 6 **Sep 12th to 16th** FYI – Pre-Exam Days	**Module: Programming with C, continued** **PSET Mario:** Students further their understanding of loops and their familiarity with the syntax of C, by creating a program that outputs the famous Mario pyramid using spaces and hashes. **PSET Credit:** Students will put the concepts from C together, using loops, iteration, booleans, and data types to implement a program that prompts the user for a credit card number and then reports (via printf) whether it is a valid American Express, MasterCard, or Visa card number, per the definitions of each's format.

Week / Date	Topic / Projects / Assessments
Week 7 **Sep 19th to 23rd**	**Module: Arrays** **Lecture: Arrays** In this module, students will expand upon their knowledge of the fundamentals of computer programming and begin building abstractions of their own. They'll also learn about strategies for debugging their own programs. **PSET Old Friends:** Students begin to interact with their programs at the command line, allowing them to run differently each time, instead of always doing the same thing. Here, students modify some of the previous problems to allow them to be run from the command line.
Week 8 **Sep 26th to 30th** **2 Days of Class** *28-30 ~Teacher's Conference*	**Q1 Final Exam** **PSET Calc:** Students continue to interact with their programs at the command line. They will implement the basic features of calculators including addition, subtraction, multiplication, division, and modulo.
Week 9 **Oct 3rd to 7th** **3 Days of Class** *6-7 ~Q1 Exams*	**Module: Arrays, continued** **PSET Calc, continued.**

2nd QUARTER – TENTATIVE COURSE CONTENT

(NB: Depending on time and interest, the teacher may delete and/or add other selections.)	
Week / Date	Topic / Projects / Assessments
Week 1 (10) **Oct 10th to 14th** **4 Days of Class** *10 – Double 10 Holiday*	**Module: Arrays, continued** **PSET Caesar:** Students dive into cryptography—the transformation of "plaintexts" to instead be secret messages, and how we can use machines to do this for us. Here they implement their own version of a caesar cipher.
Week 2 (11) **Oct 17th to 21st**	**Module: Arrays, continued** **PSET Vigenére:** Furthering their understanding of cryptography, students will create a cipher more secure than a caesar cipher, a vigenére cipher, where a keyword is used to encrypt the message.
Week 3 (12) **Oct 24th to 28th** *25-27 – Book Fair* *28- Masquerade Night* *TBA-Holy Rosary Mass*	**Module: Arrays, continued** **PSET Crack:** After learning about encrypting "plaintext," students will explore the opposite. They will create a program, using varying levels of abstraction such as functions and libraries to help them decrypt encrypted passwords.
Week 4 (13) **Oct 31st to Nov 4th** *1-All Saint's Day Mass*	**Module: Algorithms**

	Algorithms: lecture **Lab Sort:** Students will analyze three sorting programs to determine which algorithms they use. **PSET Fifteen:** Students take their newfound knowledge of functions and organizing data and aided by some distribution code that implements the basic framework for them, to collaborate on implementing the classic Game of Fifteen with user-interactivity, while explaining their implementations with other groups.
Week 5 (14) **Nov 7ᵗʰ to 11ᵗʰ**	**Module: Algorithms, continued** **PSET Plurality:** Students will implement a program that runs a plurality election, in which every voter gets to vote for one candidate. At the end of the election, whichever candidate has the greatest number of votes is declared the winner of the election. **PSET Runoff:** implement a program that runs a runoff election where voters can rank as many candidates as they wish. If no candidate has more than 50% of the vote, then an "instant runoff" occurs. The candidate who received the fewest number of votes is eliminated from the election, and anyone who originally chose that candidate as their first preference now has their second preference considered.
Week 6 (15) **Nov 14ᵗʰ to 18ᵗʰ**	**Module: Memory** **Lecture: Memory** Students will uncover some of the concepts that go on under the hood when we use different data structures like strings and arrays. Then students explore ways to store data of various types in a struct. **Lab Volume:** Students will write a program to modify the volume of an audio file. **PSET Whodunit:** Students explore images in depth and the varying levels of abstraction used to represent an image, rooting back to the individual bits that compose the pixels within an image. They will both individually and in teams, modify bitmap images to extract a hidden image. Additionally, they will answer some questions about images more generally.
Week 7 (16) **Nov 21ˢᵗ to 25ᵗʰ** *25 - YSC Contest* *25-Gr.12 Q2 Exam*	**Module: Memory Quiz** **PSET Filter:** Students will implement a program that applies filters to BMPs, reflecting, blurring, or sepia. **Q2 Final Exam**

Week / Date	Topic / Projects / Assessments
Week 8 (17) **Nov 28th to Dec 2nd** FYI – Pre-Exam Days *28-Gr.12 Q2 Exam*	**PSET Recover:** In this problem, students will receive the file of a corrupted memory card storing 50 jpegs. They will work in groups to use their knowledge of file I/O to read the images from the memory card and write them to new files, thus restoring the lost images.
Week 9 (18) **Dec 5th to 9th** 8 - Foundation Day Celebrations	AP Create Task Prep
Week 10 (19) **Dec 12th to 16th** **3 Days of Class** *15-16 ~Q2 Exams*	AP Create Task Prep
Dec 19th to Jan 2nd	**Christmas Break**

3rd QUARTER – TENTATIVE COURSE CONTENT

(NB: Depending on time and interest, the teacher may delete and/or add other selections.)	
Week / Date	Topic / Projects / Assessments
Week 1 (20) **Jan 5 to 6th** **2 Days of Class**	**AP Create task Time** All Classes provided
Week 2 (21) **Jan 9th to 13th**	**AP Create task Time** All Classes provided
Week 3 (22) **Jan 16th to 20th**	**AP Create task Time** Four Classes provided **Module: Data Structures** **Lecture: Data Structures** As students begin to wrap up their time in C, they are challenged to consider more complex data structures. Students dive into the various pros and cons of the various data structures and which are better to use in various scenarios.
Jan 23rd to 27th	**Chinese New Year**
Week 4 (23) **Jan 30th to Feb 3rd**	**Module: Data Structures, continued** **Lab Inheritance:** Students simulate the inheritance of blood types for each member of a family. **PSET Speller:** Students use their new-found knowledge of data structures to implement a spell checker in C. They implement several functions that work within staff-provided distribution code and test their code with various text files.

	Data Structures Quiz
Week 5 (24) **Feb 6th to 10th**	**Module: Python** **Lecture: Python** Students build upon their knowledge gained in the course to learn several new programming languages with abstractions built in that allow them to go far beyond what simply C and Scratch are able to do. They solve more complex problems that require processing large amounts of data and dealing with processes that scale and see how these techniques can be applied to confront the challenges computer scientists will be contending with in the future. **Lab World Cup:** Students write a program to run simulations of the FIFA World Cup. **PSET Hello:** Students rewrite Hello in Python **PSET Mario:** Students rewrite Mario in Python **PSET Cash:** Students rewrite Credit in Python **PSET Credit:** Students rewrite Credit in Python
Week 6 (25) **Feb 13th to 17th**	**Module: Python, continued** **PSET Readability:** Students implement a program that computes the approximate grade level needed to comprehend some text according to the Coleman-Liau formula. **PSET DNA:** Students implement a program that identifies a person based on their DNA, by writing a program that will take a sequence of DNA and a CSV file containing STR counts for a list of individuals and then output to whom the DNA (most likely) belongs. **Python Quiz**
Week 7 (26) **Feb 20th to 24th** *20-24 ~IOWA* *22 ~ Ash Wednesday Mass* *21-23 ~ Pre-Exam Days*	**Module SQL** **Lecture SQL** Students learn about how applications store data in databases. They solve problems that require processing large amounts of data and dealing with processes that scale and see how these techniques can be applied to confront the challenges computer scientists will be contending with in the future. **Lab Songs:** Students write SQL queries to answer questions about a database of songs.
Week 8 (27) **Feb 27th to March3rd** **3 Days of Class** 27-28 ~ 228 Memorial Day Holiday	**PSET Movies:** Students will write SQL queries to answer questions about a database of movies. **SQL Quiz**
Week 9 (28) **March 6th to 10th** **4 Days of Class**	**PSET Fiftyville:** Students will write SQL queries to solve a mystery.

11 – Q3 Exams	**Q3 Final Exam**

4th QUARTER – TENTATIVE COURSE CONTENT

(NB: Depending on time and interest, the teacher may delete and/or add other selections.)	
Week / Date	Topic / Projects / Assessments
Week 1 (29) **March 13th to 17th** **4 Days of Class** *13 – Q3 Exams* *14~ Q4 Begins*	**Module HTML, CSS, JavaScript** **Lecture HTML, CSS, JavaScript** Students learn about the internet, the web and how HTML, CSS and Javascript create interactive web pages. **Lab Trivia:** Students will write a webpage that lets users answer trivia questions.
Week 2 (30) **March 20th to 24th** *20 ~ Fire Drill*	**Module HTML, CSS, JavaScript, Continued** **PSET Homepage:** Students will build a simple homepage using HTML, CSS, and JavaScript.
Week 3 (31) **March 27th to 31st**	**Module Flask** **Lecture Flask** **Lab Birthdays:** Students will Create a web application to keep track of friends' birthdays. **PSET Finance:** Implement a website via which users can "buy" and "sell" stocks, using an API to fetch prices.
Apr 3rd to 14th	**Easter Break**
Week 4 (33) **Apr 17th to 21st**	**PSET Finance, continued** **Module Security** **Lecture Security** Defender of the Web: Students explore the notions of cyberattacks and cybersecurity and investigate in more detail some of the common types of attacks that impact websites today.
Week 5 (34) **Apr 24th to 28th** *24-28 ~ AP Mock Exams*	Mock Exams **Module Impact of Computing** **Simulate! (Simulation):** In this writing problem, students research a computer simulation of their choice. They will explain how the simulation works and the benefits and disadvantages of using it. Does the program account for all the features it is trying to model? Does the model rely on any assumptions? Are there downsides to using a program instead of testing in the real-world? What are those downsides?
Week 6 (35)	AP Exams

May 1st to 5th *2-4~ Pre-Exam* *1-5~ Final Exams (K, 5, 8, 12 only)* *1-5 ~ AP Exams*	Final Project
Week 7 (36) **May 8th to 12th** *8-12~ Final Exams(K, 5, 8, 12 only)* *1-5 ~ AP Exams*	Final Exams Final Project
Week 8 (37) **May 15th to 19th** **3 Days of Class** *18-19~ Q4 Exams*	Final Exams Final Project
Week 9 (38) **May 22nd to 26th** **4 Days of Class** *22~ Record Day* *23-26 ~ Student Clearance*	Clearance Final Project
Week 10 (39) **May 29th to June 2nd** **4 Days of Class** *1 ~ Students Last Day* *2~ Teachers/Staff Meeting*	Final Project Sharing