AP COMPUTER SCIENCE PRINCIPLES

COURSE SYLLABUS

SWEETWATER HIGH SCHOOL

INSTRUCTOR: ART LOPEZ

CSE 3 COMPUTER SCIENCE COURSE

UNIVERSITY OF CALIFORNIA, SAN DIEGO

DR. BETH SIMON

Contact Information

ROOM LOCATION:	Main Building, Room 215
SWEETWATER HIGH SCHOOL PHONE NUMBER AND EXTENSION:	619.474.9700 Extension 38215
E-MAIL ADDRESSES:	Parents: arthur.lopez@sweetwaterschools.org Period 1: pd.1.mr.lopez@gmail.com Period 2: pd.2.mr.lopez@gmail.com Period 5: pd.5.mr.lopez@gmail.com Period 6: pd.6.mr.lopez@gmail.com
SWEETWATER CANVAS PROGRAM WEB SITE:	https://sweetwaterschools.instructure.com/login
AFTER SCHOOL TUTORING HOURS:	Varies from week to week
SWEETWATER HIGH SCHOOL WEB ADDRESS	http://suh.sweetwaterschools.org/

Course Description

"AP Computer Science Principles introduces students to the foundational concepts of computer science and challenges them to explore how computing and technology can impact the world. With a unique focus on creative problem solving and real-world applications, AP Computer Science Principles prepares students for college and career." *College Board AP Computer Science Principles Description*

What is this course about?

"You have used computers your whole life. Mostly you don't even think about it. Your phone has several computers in it, your iPad, your Kindle, your TV, your car has several in it unless it is an antique, and on and on. Most devices with an on/off switch contain computers these days. And they continue to be more widely used – planes fly without pilots, vehicles can drive on highways autonomously, social media like Facebook and Twitter add features every month that allow new forms of interaction.

That list probably contains no surprises to you – it describes how the world is. You're doing just

fine with today's technology. Luckily, this class isn't about how the world is. The world will change. This class is about what you need to know for the world that hasn't yet arrived, and which you will create – regardless of your chosen major or intended career." *Dr. Beth Simon, PHD Computer Science, UC-San Diego*

What does this course offer to you?

In this course, you will gain the basic level of understanding computers and computation that we think **ALL** college and high school graduates should have – to be prepared to create the future. Last century, competence was defined by the three Rs: reading, writing and arithmetic. They are required to pursue a professional career in any discipline. This is the 21st century, and it's clear that computing is poised to permeate not only our professional work, but also our society. So think of understanding and skills of computing as a fourth 'R', necessary for any discipline. After this class, you will stand out from other college and high school graduates in your preparation to work with new technologies of the future.

Students who previously took the course described some of the following things they got out of it:

- **Confidence**: "It has given me confidence that I'm able to figure things out on a computer that I never would have thought that I could do."
- **View of Technology**: "Now, every time I find myself playing a video game, I actually understand what makes it work. That these games are not magically produced, that it takes time, skill, and sufficient funds to create these games. I appreciate these games more than before taking this class."
- **Analysis Skills**: "Programming allows a person to think more logically, thinking in order and debugging allows the user to gain valuable problem solving skills. Aspiring to go to law school, thinking logically is extremely important and I think this has helped."
- **Communication Skills**: "In today's technologically-centered world, using a program like ALICE gives us valuable exposure to discussing things technically with other people and explaining clearly what we are trying to do."
- **Organizational Skills**: "Through ALICE, I learned to stay organized and structured in anything I do, including studying for other classes. Although at first, thinking with several concepts at a time was very difficult, now I am more confident."

Dr. Beth Simon, PHD Computer Science, UC-San Diego

How will you (and the teacher) know if you are making progress in your learning (Assessments)?

The course offers you a number of opportunities to get feedback on whether you are learning what you need to know; learning computing and computational thinking is not done by reading about it. Moreover, we care more about your development of analysis and communication skills regarding computing and computational thinking than whether you can "just do it". For each topic, you will be able to get practice and feedback in the following ways:

- **Exploratory modules and projects with on-line book:** You will complete exploratory projects based on modules that guide you in understanding a topic by creating programs on the computer to get the basics down.
- **Guided Practice Assignments**: The instructor, your classmates, and you will work on creating projects together in the computer lab based on the topic you are learning. This will reinforce the computing principle you are learning from the exploratory modules in the on-line book.
- **Independent Assignments Lab**: You will be expected to demonstrate your mastery of the concepts from the on-line book modules by completing labs and projects. You are expected to come prepared to lab -- having done the relevant exploratory projects and assignments and having engaged with and learned the material in lecture and on-line modules. You are expected to complete your lab during the session, but you will be able to get assistance from not only tutors, but your fellow students. Labs and projects will be posted in advance and you are encouraged to look over them before you come to lab.
- **Quizzes (Clicker Questions):** At the beginning of lecture, you will take a quiz answering a few basic questions from the homework – this is your incentive read the on-line book to do the exploratory projects and provide feedback to you on whether you are ready to attend lecture.
- **Discussion Peer Instruction Questions (Clicker Questions):** During lecture, you will deepen your understanding of computing concepts and develop technical analysis and communication skills by discussing challenging questions in a team of three students, guided by the instructor.
- **Technology and Society:** You will be given various activities exploring issues on involving technology and society. These activities will require you to use discussion forums, Internet resource finding/analysis, and wiki development.
- **Performance Tasks:** The through-course component is comprised of two performance tasks — separately, these tasks require students to engage with ***creating a computer program*** and ***exploring a computer innovation***.
- **Midterm and Final**: a midterm and final will be given in order for you to demonstrate mastery of the topics and concepts presented in class.
- **AP EXAM**.

Online Resources

Instead of a traditional, static, "textbook" -- in this course we provide all necessary materials online. You will be asked to read and complete activities using these materials for each class. You can read more about this in the "How to Do Exploratory Homework" section.

- On-line textbook:
  - Expeditions through ALICE: https://sites.google.com/a/eng.ucsd.edu/expeditions-through-ALICE/home
- Alice (Version 2.4) [Software]. Carnegie Mellon University, 1999. http://www.alice.org/index.php?page5downloads/download_alice2.4.
- Computer Science Unplugged. Michael Fellows, Tim Bell, and Ian Witten. http://csunplugged.org/activities/.
- Pencilcode.net: Web-based programming platform.
- "Repository for Alice Materials." Susan Rodger. Duke University Department of Computer Science. http://www.cs.duke.edu/csed/alice09/.
- Abelson, Hal, Ken Ledeen, and Harry Lewis. Blown to Bits: Your Life, Liberty, and Happiness after the Digital Explosion. Addison-Wesley, 2008. http://www.bitsbook.com/thebook/.
- "Pair Programming-in-a-Box: The Power of Collaborative Learning." National Center for Women & Information Technology. https://www.ncwit.org/pair-programming.

Supplemental Textbook:

- Dann, Wanda P., Stephen Cooper, and Randy Pausch. *Learning to Program with Alice, 3rd Edition*. Boston: Prentice Hall, 2012.

What will this course teach you?

The AP Computer Science Principles course taught at Sweetwater High School is based upon:

- The College Board's AP Computer Science Principles Curriculum Framework.
- The College Board's Computer Science Principles Course Planning Guide 1 written by Art Lopez.
- The curriculum taught for this course is created, designed and provided by Dr. Beth Simon of UC-San Diego.
    - This same curriculum is taught for CSE3 course at UCSD (University of California San Diego) and SDSU (San Diego State University) CS 100.

The course is engaging, rigorous and rich in computational content, includes computational and critical thinking skills, and engages students in the creative aspects of the field. Through both its content and pedagogy, this course aims to appeal to a broad audience.

The students are required to learn the following ***seven big ideas*** and ***computational thinking practices*** based upon the College Board's AP CSP Curriculum Framework:

SEVEN BIG IDEAS

- Big Idea 1: Creativity.
- Big Idea 2: Abstraction.
- Big Idea 3: Data and Information.
- Big Idea 4: Algorithms.
- Big Idea 5: Programming.
- Big Idea 6: The Internet.
- Big Idea 7: Global Impact.

The units that follow interweave the six Computer Science Principles Computational thinking practices listed below:

COMPUTATIONAL THINKING PRACTICES

- [P1] Connecting computing
- [P2] Creating computational artifacts
- [P3] Abstracting
- [P4] Analyzing problems and artifacts
- [P5] Communicating
- [P6] Collaborating

College Board Assessments: Computer Science Principles Performance Tasks and AP Exam

The learning objectives will be the targets of assessment for the AP Computer Science Principles course. This assessment comprises two parts: the end of course AP Exam and the through course AP assessment comprised of two performance tasks.

- The **AP Computer Science Principles** exam will be a multiple choice, paper and pencil exam in which students will demonstrate achievement of the course objectives.
- The **Performance Task Explore**, which requires students to explore the impacts of computing.
- The **Performance Task Create**, which requires students to create computational artifacts through programming.

Schedule of Topics

By the end of this course, you should master the ***computer science principles*** taught in this course in order to gain a thorough understanding of how computers work (conditional control flow, iteration, for example) and to gain appreciation for the value of abstraction (e.g., methods, parameters).

Semester 1 Units	*Guiding Questions*	*College Board Big Ideas, Essential Understandings and Learning Objectives*
❖ **Unit 1: Introduction to Computer Science Principles** ➢ Using LMS: Canvas. ➢ Defining, describing or explaining Computer Science. ➢ Defining, describing or explaining Computational Thinking. ➢ Describing or explaining purpose of Pair Programming methods and strategies for learning. ➢ Module 1: Telling a Story. ▪ Introduction to Alice Programming Platform. ▪ Defining computing and computational thinking. ➢ Creating a program. ▪ Defining/Describing/Explaining a computer program. ▪ Defining/Describing/Explaining computer programs as a concept of an algorithm. ▪ Exploring. ▪ Planning and/or Storyboarding. ▪ Creating. ▪ Predicting. ▪ Testing. ➢ The Internet as a Social Experience ➢ Global impact of social media	➢ What are computing and computational thinking and why are the important to learn? ➢ What is Computer Programming? ➢ Why is creating a plan important for creating a program? ➢ Why do experienced programmers use prediction when creating and writing programs?	**Big Ideas**: Creativity, Algorithms, Programming, Global Impact **EU's**: 1.1, 1.2, 2.2, 4.1, 4.2, 5.1, 5.2, 5.4, 7.1, 7.2, 7.4 **LO's**: 1.1.1, 1.2.1, 1.2.2, 1.2.3, 1.2.4, 1.2.5, 1.3.1, 2.3.1, 4.1.1, 4.1.2, 4.2.4, 5.1.1, 5.1.2, 5.2.1, 5.4.1, 7.1.1, 7.3.1, 7.4.1

❖ **Unit 2: Telling and Dividing a Story** ➢ Module 2: Dividing the Story. ➢ Module 3: Stories from Pieces. (the following refer to both Modules 2 and 3) ▪ Defining methods (procedures). ▪ Advantages of using methods within a program. ▪ Creating methods in a program. ▪ Methods applied as a concept of abstraction.	➢ What are the advantages of using methods/procedures within computer programs? ➢ Why is abstraction a key concept for programmers to understand and use in computer programs? ➢ How do we use abstraction in our daily lives? ➢ How are methods/procedures applied as a concept of abstraction in computing and computer programs.	**Big Ideas**: Creativity, Abstraction, Algorithms, Programming, Global Impact **EU's**: 1.1, 1.2, 1.3, 2.2, 2.3, 4.1, 4.2, 5.1, 5.2, 5.3, 5.4 **LO's**: 1.1.1, 1.2.1, 1.2.2, 1.2.3, 1.2.4, 1.2.5, 1.3.1, 2.2.1, 2.2.2, 2.2.3, 2.3.1, 4.1.1, 4.1.2, 4.2.4, 5.1.1, 5.1.2, 5.1.3, 5.2.1, 5.3.1, 5.4.1
❖ **Unit 3: Acting the Same and Acting Differently (Parameters**) ➢ Module 4: Acting the Same ➢ Module 5: Acting Differently ▪ Define/Describe/Explain Parameters. ▪ Advantages of using parameters in computer programs. ▪ Parameters applied as a concept of abstraction. ➢ Technology and Society 1 ▪ Quality of Information from the Web. ▪ Analysis of Risks of Technology.	➢ What are parameters and how are they used in computer program? ➢ What are the advantages of using parameters within computer programs? ➢ How are parameters applied as a concept of abstraction in computing and computer programs? ➢ How are parameters used in our daily lives? What are some examples?	**Big Ideas**: Creativity, Abstraction, Algorithms, Programming, Global Impact **EU's**: 1.1, 1.2, 1.3, 2.2, 2.3, 4.1, 5.1, 5.2, 5.3, 5.4, 5.5, 7.1, 7.2, 7.3, 7.5 **LO's**: 1.1.1, 1.2.1, 1.2.2, 1.2.3, 1.2.4, 1.2.5, 1.3.1, 2.2.1, 2.2.2, 2.2.3, 2.3.1, 4.1.1, 4.1.2, 5.1.1, 5.1.2, 5.1.3, 5.2.1, 5.3.1, 5.4.1, 5.5.1, 7.1.1, 7.1.2, 7.2.1, 7.3.1, 7.5.2

❖ **Unit 4: Getting Into the Story (Events)** ➢ Module 6: Getting Into the Story (Events) ▪ Defining/Describing/Explaining Events. ▪ Making computer programs interactive with Events. ▪ Different types of Events. ▪ Defining/Describing/Explaining Event Handlers. ▪ Rewrite code so that it can be reused to handle more than one event.	➢ What are events and how are they used in interactive programs? ➢ How would you describe the different kinds of events that enable a user to interact with computer programs in different ways? ➢ What is an event handler and how does the event handler allow events to call methods? ➢ How can you write code so that it is reused to handle more than one event?	**Big Ideas**: Creativity, Abstraction, Algorithms, Programming **EU's**: 1.1, 1.2, 1.3, 2.2, 2.3, 4.1, 5.1, 5.2, 5.3, 5.4 **LO's**: 1.1.1, 1.2.1, 1.2.2, 1.2.3, 1.2.4, 1.2.5, 1.3.1, 2.2.1, 2.3.1, 2.3.2, 4.1.1, 4.1.2, 5.1.1, 5.1.2, 5.1.3, 5.2.1, 5.3.1, 5.4.1
❖ **Unit 5: The Internet and Binary Numbers** ❖ **Technology and Society 2:** ➢ **Digital Responsibility: Understanding the Internet Infrastructure**	➢ How would you explain the Internet in everyday language? ➢ How has the Internet shaped our current society and how may they change the future? ➢ What are the pro and cons of encryption versus free speech?	**Big Ideas**: Abstraction, Algorithms, The Internet **EU's**: 2.1, 2.2, 2.3, 3.1, 4.2, 6.1, 6.2, 6.3, 7.5 **LO's**: 2.1.1, 2.1.2, 4.2.1, 6.1.1, 6.2.1, 6.2.2, 6.3.1, 7.5.1
❖ **AP Performance Task Explore** ➢ **Impact of Computing Innovations**		
❖ **Final Exam Semester 1**		

Semester 2		
❖ **Unit 6: Calculating Realism (Mathematical Expressions and Functions)** ➢ Module 7: Calculating Realism (Mathematical Expressions and Functions). ▪ Defining/Describing/Explaining **Functions**. ▪ Defining/Describing/Explaining **Algorithms**. ▪ Differences between a function and a method. ▪ Using mathematical expressions and functions in a computer program. ▪ Using functions to **abstract** complex mathematical calculations.	➢ What does a function do within a computer program? ➢ What is the difference between a function and a method? ➢ What are mathematical expressions and functions and how are they used in computer programs? ➢ How can functions be used to abstract complex mathematical calculations?	**Big Ideas**: Creativity, Abstraction, Algorithms, Programming **EU's**: 1.1, 1.2, 1.3, 2.2, 2.3, 4.1, 5.1, 5.2, 5.3, 5.4 **LO's**: 1.1.1, 1.2.1, 1.2.2, 1.2.3, 1.2.4, 1.2.5, 1.3.1, 2.2.1, 2.3.1, 2.3.2, 4.1.1, 4.1.2, 5.1.1, 5.1.2, 5.1.3, 5.2.1, 5.3.1, 5.4.1
❖ **Unit 7: Choosing Your Path (If-Else statements)** ➢ Module 8 Choosing Your Path (If Statements) ▪ Defining/Describing/Explaining conditional or "If-Else" statements. ▪ Defining/Describing/Explaining conditional behaviors. ▪ Defining/Describing/Explaining If-else statements as decision points in a computer program. ❖ **Technology and Society 3:** ➢ **If 's and Events On the Web**	➢ What are If-Else statements and how are they used in computer programs? ➢ What are conditional behaviors? What are some examples used in computer programs and in your daily life? ➢ How are If-Else statements related to decisions and decision points in computer programs? ➢ What can If-Else statements evaluate when making decisions within programs?	**Big Ideas**: Creativity, Abstraction, Algorithms, Programming **EU's**: 1.1, 1.2, 1.3, 2.2, 2.3, 4.1, 5.1, 5.2, 5.3, 5.4, 5.5 **LO's**: 1.1.1, 1.2.1, 1.2.2, 1.2.3, 1.2.4, 1.2.5, 1.3.1, 2.2.1, 2.3.1, 4.1.1, 4.1.2, 5.1.1, 5.1.2, 5.1.3, 5.2.1, 5.3.1, 5.4.1, 5.5.1

❖ **Unit 8: More Complex Control of Execution (Compound Boolean Expressions and Nested Conditional or If-Else Statements)** ➢ Module 9 More Complex Control of Execution (Compound Boolean Expressions and Nested If-Else statements) ▪ Defining/Describing/Explaining Boolean Expressions. ▪ Defining/Describing/Explaining complex sets of conditions in computer programs. ▪ Defining/Describing/Explaining nested If-Else statements. ▪ Creating computer programs where execution is controlled by complex sets of conditions. ❖ **Technology and Society Assignment 4:** ➢ Searching for Meaningful Information: Advanced Google Search and Google Scholar	➢ How are programs created where execution is controlled by complex sets of conditions? ➢ What are nested If-Else statements and compound Boolean expressions? ➢ Can you explain the complex sets of conditions that would require the use of a compound Boolean expressions within a If-Else statement or nested If-Else statement? Give examples.	**Big Ideas**: Creativity, Abstraction, Algorithms, Programming **EU's**: 1.1, 1.2, 1.3, 2.2, 2.3, 4.1, 5.1, 5.2, 5.3, 5.4, 5.5 **LO's**: 1.1.1, 1.2.1, 1.2.2, 1.2.3, 1.2.4, 1.2.5, 1.3.1, 2.2.1, 2.3.1, 4.1.1, 4.1.2, 5.1.1, 5.1.2, 5.1.3, 5.2.1, 5.3.1, 5.4.1, 5.5.1

❖ **Unit 9: Doing Things Over and Over (Loops)** ➢ Module 10: Doing Things Over and Over (Loops). ▪ Defining/Describing/Explaining Iterations. ▪ Defining/Describing/Explaining Loops as a concept of iteration ▪ Defining/Describing/Explaining counted and nested loops and how they are used in programs. ▪ Defining/Describing/Explaining conditional "while" loops and Boolean expressions and how they are used to execute a set of computing instructions.	➢ What are iterations, and how are loops associated with iterations? ➢ What are "counted" and "nested" loops, and how are they used in computer programs? What are some examples? ➢ How are nested loops used in computer programs to create more complex behaviors or make repeated things happen in more complicated ways? ➢ How are "conditional (while)" loops and Boolean expressions used to execute a set of computing intstructions?	**Big Ideas**: Creativity, Abstraction, Algorithms, Programming **EU's**: 1.1, 1.2, 1.3, 2.2, 4.1, 5.1, 5.2, 5.3, 5.4, 5.5 **LO's**: 1.1.1, 1.2.1, 1.2.2, 1.2.3, 1.2.4, 1.2.5, 1.3.1, 2.2.1, 2.3.1, 4.1.1, 4.1.2, 5.1.1, 5.1.2, 5.1.3, 5.2.1, 5.3.1, 5.4.1, 5.5.1
❖ **Unit 10: Grouping Items Together (Lists or Arrays)** ➢ Module 11: Grouping Items Together (Lists). ▪ Defining/Describing/Explaining Lists or Arrays. ▪ Defining/Describing/Explaining Randomness. ▪ Using randomness within lists. ▪ Using loops with lists and randomness to make computer programs more interesting.	➢ What are lists (also known as arrays) and what do they allow to be done in computer programs? ➢ How does randomness allow a group of objects to perform the same action but do it slightly differently within lists? What are some examples? ➢ How can looping be used with lists and randomness to make computer programs more interesting to use? What are some examples	**Big Ideas**: Creativity, Abstraction, Algorithms, Programming **EU's**: 1.1, 1.2, 1.3, 2.2, 2.3, 4.1, 5.1, 5.2, 5.3, 5.4, 5.5 **LO's**: 1.1.1, 1.2.1, 1.2.2, 1.2.3, 1.2.4, 1.2.5, 1.3.1, 2.2.1, 2.3.1, 2.3.2, 4.1.1, 4.1.2, 5.1.1, 5.1.2, 5.1.3, 5.2.1, 5.3.1, 5.4.1, 5.5.1
❖ **AP Performance Task Create:** ➢ **Create – Applications from Ideas**		

❖ **Unit 11: Intro to Spreadsheets and Working with Large Data Sets** ➢ Module 12: Intro to Spreadsheets ➢ Module 13: Spreadsheets – Working with Large Data Sets (the following refers to both Modules 12 and 13) ▪ Defining/Describing/Explaining Spreadsheets. ▪ Applying Computational Concepts to Computer Applications such as Excel or Google Sheets. ▪ Applying computer terminology to Computer Applications such as Excel or Google Sheets. ▪ Working with large data sets in spreadsheet applications. ▪ Defining/Describing/Explaining why large data sets are important in computing and computational thinking.	➢ How can you apply the concepts or programming that you learned with Alice to a computer application such as Excel or Google Sheets? ➢ What are the similarities between the terminology (i.e. parameters) and programming concepts (i.e. functions) for creating programs in Alice and spreadsheets in a spreadsheet program? ➢ What are some of the basic functions of spreadsheet programs that will increase a computer user's productivity in using those types of programs? Why are large data sets important in computing and computational thinking?	**Big Ideas**: Creativity, Abstraction, Data and Information Algorithms, Programming **EU's**: 1.1, 1.2, 2.2, 3.1, 3.2, 3.3, 4.1, 5.1, 5.2, 5.3, 5.4, 5.5 **LO's**: 1.1.1, 1.2.1, 1.2.2, 1.2.5, 2.2.1, 3.3.1, 3.1.2, 3.1.3, 3.2.1, 3.2.2, 3.3.1, 4.1.1, 4.1.2, 5.1.1, 5.1.2, 5.1.3, 5.2.1, 5.3.1, 5.4.1, 5.5.1
Final Exam		

Grading Policy

Students' academic grades will be based on the following an available on the Canvas course:

Categories	Percentage of Grade
Clicker Quizzes	5%
Clicker Discussion Questions	5%
Module Questions	5%
Technology and Society	5%
Labs	15%

Programming Exercises	10%
Performance Task Project	15%
Midterm	15%
Final	25%

The grading scale is as follows:

Letter Grade	Percentage	Meaning
A	100 – 90%	Student has Mastered standards.
B	89 – 80%	Student Exceeds standards.
C	79 – 70%	Student Meets standards.
D	69% - 60%	Student is below the expected standards.
Fail	Below 59%	Student does not meet standards.

Missing or Turning Assignments in Late

Missing or turning in assignments late have severe consequences on your grade.

Please, to the best of your ability, do not miss assignments. If you miss an assignment, you are allowed to do the following.

Turn in it 1 to 2 days late: highest grade you can receive is 70%.

Turn in it 3 to 4 days late: highest grade you can receive is 60%.

After the 4th day, the highest grade you can receive is a 50%.

Redoing Assignments

You can redo some assignments and only if you got a D or an F grade. The highest grade you can receive is 70% or a C-, ***if you do it correctly***. The instructor will decide which assignments can be redone.

You have up to one week to redo the assignment. You can only redo the assignment ***after school or at lunchtime if the room is open***.

Take responsibility for yourself. It is up to you to make up or redo assignments! It is, after all, your grade.

Citizenship Grades and Attendance Policy

Students' citizenship grades will be based on the following:

- Attendance.
- Expected Student Behaviors.
- Student effort in class.
- Student materials.

The attendance policy is set forth by Sweetwater High School, their class participation is set forth in the ***Students' Responsibilities*** section of the Sweetwater High School handbook, and the student effort is set by the **scholarship** grade.

For attendance, each **block class** equals **2 school hours**. Therefore, if a student misses ***one block class***, it will count as ***two absences***.

12-hour absences, either excused or unexcused, will result in an F grade in Citizenship!

20-hour absences, either excused or unexcused, will result in an F grade in both Scholarship and Citizenship as outlined per district policy.

A combination of 5 tardies and/or truancies will result in an F grade in Citizenship!

The following tables list the highest grade possible due to being late, truancies and/or absences.

Quarter Grades

Absences (In Hours)	***Tardies***	***Truancies***
A: 0 - 2	A: 0	A: 0
B: 3	B: 1	B: 1
C: 4	C: 2	C: 2
D: 5	D: 3	D: 3
F: 6 or more.	F: 4 or more.	F: 4 or more.

Semester Grades

Absences (In Hours)	*Tardies*	*Truancies*
A: 0 - 4	A: 0 - 2	A: 0
B: 5 - 7	B: 3	B: 1
C: 8-10	C: 3	C: 1
D: 11-12	D: 4	D: 2
F: 13	F: 5 or more.	F: 3 or more.

Being Late: If you are late, it can be cleared by attending the detentions in the library or Saturday School.

Student Materials

- **USB drive** (also known as a jump, flash or thumb drive). Try to get a 128 MB drive as the minimum. This will be used to back up the files you create in this class. You can purchase this at any computer, electronics or office store.
- Student binder including a minimum of 10 clean writing sheets or a notebook.
- Blue or black ink pen, a No. 2 pencil, and a highlighter.
- Class notes (keep them dated and numbered in the class folder).
- Daily participation in class discussion.
- If *absent*, students are ***responsible*** for making up missing assignments.
- All missing assignments must be turned in 1 week after receiving assignments.
- All missing assignments are due one week prior to the end of the quarter or semester.

Best Learning Community/Environment and Expected Student Behaviors

We want to create a learning community that includes the most positive, creative, engaging, and respectful learning environment possible. Therefore, students are ***expected to obey the following rules and behave in the following manner*** in order to create this great leaning community and environment.

1. **Respect** the dignity of fellow students and the instructor. The instructor will in turn treat the student with the utmost respect. This includes the following:
    a. Please do not talk to the teacher and/or your peers in a disrespectful or rude manner.
    b. Please watch the tone of your voice. The teacher is not one of your friends! The teacher, in this course at least, is someone that deeply cares about your education. Do not talk to the instructor as if they are one of your friends at school. Talk to the teacher as an adult person.
    c. ***Appropriate body language***. When asked to do something that is required for the class, please do not do the following:
        i. Make inappropriate facial gestures (no dirty looks please!).
        ii. Do not roll your eyes.
        iii. Do not slouch in your chair. Please sit up straight.
    d. Do not slam your books or backpack on the floor.
    e. ***Do not challenge the teacher in the classroom***. If you choose to do so, you will be sent to the office.
2. No eating, drinking or chewing gum is permitted in class.
3. Grooming or applying makeup is not permitted.
4. Do not throw objects in the classroom or write on equipment.
5. Do not talk unless permission is given.
6. Do not play around the computers. You are responsible for the cost of the computer if you damage the equipment by horsing around.

Inappropriate or disruptive behavior is unacceptable in this class. Problems will first be discussed with the student. On **second** offense, a referral will be written and the student will be sent to the principal's office for further disciplinary action. At the teacher's discretion, a referral will be sent to the principal's office **immediately without a first warning if deemed necessary**.

Electronic Devices

Electronic devices are not allowed to be on during this course unless permitted by the instructor. Your education and those of your classmates are the reason you are here in school.

Sometimes having these electronic devices prevent you from doing your best in school. You are not allowed to have the following on or open unless given permission by the instructor. These items should be off and in your backpacks:

- Mobile phones.
- MP3 or CD players of any type.
- Any other electronic device I have not mentioned that is distracting to others and your own learning.

You will be asked to turn off the device only once during the semester, asked to put it away in your backpack and you will have an automatic detention.

The second time you are caught doing this, the device will be confiscated (taken away), given to the Assistant Principal, and your parent(s) will receive a phone call from me.

The third time will be a referral and suspension. Please follow these rules. They benefit your education.

Parents Responsibilities

Your participation of helping your teenager to learn the above stated objectives is critical for their success for this course and for their future academic and career choices.

The students are required to bring their materials everyday to class as we may use them on a daily basis. If the students do not bring their materials or choose to not follow the rules, then a **DETENTION** will be assigned immediately after class for at least 5 minutes. If your son or daughter does not follow the rules a second time, I will contact you immediately.

If your son/daughter is unwilling to put forth the maximum effort required in completing the assignments in class, I will be calling you by phone to inform you that your son/daughter is refusing to participate and do the best they can in class. By talking to them, I hope you can explain to your son/daughter the importance of trying to do their very best in class.

If the students refuse to do their best or to complete the assignments in class, then a detention will be assigned that same day during nutrition break, lunchtime or after school. The detention is normally from 5 to 30 minutes, depending on the students' behavior or actions.

If the student is unable to attend detention that day, we will then call home and make arrangements for the student to attend a detention after school **the following day**.

Both your daughter/son and I look forward to working with you on this exciting learning venture that will promote their success for the future.

At the end of this document, you will find a form that requires your signature. Your signature means that you agree to the learning process described here.

Please return the form through your daughter/son. It is part of their grade.

I am truly excited about the projects that are to be developed by your daughter/son. With your guidance and help, we will all provide the best learning environment for them!

I look forward to working with both you and your daughter/son!

Respectfully yours,

Arturo Lopez

AP Computer Science: Principles Instructor

Sweetwater High School

National City, CA

PLEASE READ, SIGN, AND RETURN THIS PAGE <u>ONLY</u> TO MR. LOPEZ BY _____ (for your first graded assignment). Students who fail to turn in this acknowledgement will receive a call to parents.

I have read and understand Mr. Lopez's **AP Computer Science Principles Syllabus and Class Requirements**. I further understand a passing grade of at least 50% is needed for the semester in order to receive full credits for the course (both semester 1 and 2).

_____ _____
Printed Student Name Signature of Student:

Period: ____. Date: _____.

Student E-mail address: _____.

<u>For Parents and Guardians</u>:

I have read and understand Mr. Lopez's syllabus and class requirements. I know that I may call Mr. Lopez at (619) 474-9700, extension 38215 or reach him via email: (arthur.lopez@sweetwaterschools.org) with concerns regarding my student in his class, and he will contact me so we may discuss those concerns.

_____ _____
Printed Parent Name Signature of Parent:

Date: _____.

Home Phone: _____.

Mobile Phone: _____.

Work Phone: _____. Extension: _____.

Parent E-mail address: _____.

**** Please complete the Parent/Student Information Form on the class website to provide me w/ preferred method of contact and any additional information you feel I may need to know about your daughter/son.***