## Teacher Contact: Janice_Bailey@mcpsmd.org
## Google Classroom Class Code: ay9hn7
## Website http://janicedbailey.com

## COURSE DESCRIPTION

AP Computer Science Principles (AP CSP) introduces you to the foundations of computer science with a focus on how computing powers the world. Along with the fundamentals of computing, you will learn to analyze data, create technology that has a practical impact, and gain a broader understanding of how computer science impacts people and society.

## Policy Regarding Plagiarism and Falsification or Fabrication of Information:

A student who fails to acknowledge (i.e., through citation, through attribution, by reference, and/or through acknowledgment in a bibliographic entry) the source or author of any and all information or evidence taken from the work of someone else will receive a score of 0 on that particular component.

Citing Sources: Students should properly cite sources that are used to provide evidence in their written responses, especially when quoting directly from a source.

Citing Program Code: Students should properly cite program code they did not explicitly write, such as APIs, open-source program code, or program code provided during peer-to-peer collaboration. Open source refers to access and licensing of program code that is publicly available for use or modification. One way students can cite someone else's program code is by adding a comment either to the program code or to the documentation, crediting the author and listing where the program code was found.

## Assessment Overview:

The AP Assessment consists of a 74-question multiple choice exam and two "through-course" assessments called the AP Performance Tasks (PTs). The tasks can be found in the official AP CS Principles Exam and Course Description

- Create Performance Task (p. 108)
- Explore Performance Task (p. 111)

## Grading

All assignments are given equal weight.
Practice Performance Tasks will make up about 25% of a student's grade.
Unit Tests will make up about 25% of a student's grade.
The remaining 50% of a student's grade will be based on their AP CSP Notebook, quizzes and other classwork.

## Late Assignments Policy

1. Homework must be complete and turned in by the due date, prior to the beginning of class. Late or incomplete homework is not accepted.
2. Late classwork will be marked down 10%. You can submit late classwork not more than two weeks past the due date, without extenuating circumstances.

Required Materials / Supplies: 1” notebook with 6 dividers and 200 sheets of lined paper. This notebook can stay in the classroom or go home with the students, but cannot be shared with another class because it will need to be turned in for grading every few weeks.

Text: Blown to Bits http://www.bitsbook.com/

Schedule at a glance:

**Digital Information**
Students will be able to use appropriate terminology to describe and compare the size of familiar digital media. Students will be able to describe how digital information is encoded, represented and manipulated. Students will be able to convert between decimal and binary numbers. Students will work in a group to produce an SVG logo.
**Resources**: Units1 and 2 on code.org: https://studio.code.org/courses/csp-2018, *Blown to Bits* Chapter 3 http://www.bitsbook.com/excerpts/chapter3/, Nick Parlante's Stanford CS101 course at https://web.stanford.edu/class/cs101/bits-gigabytes.html, SVG collection at https://codepen.io/collection/DVbvQJ/

**Introduction to Programming**
Students will be able to write basic searching and sorting algorithms. Students will be able to write functions, use parameters and write loops in the javascript language.
**Resources:** Introduction to Programming with Karel: https://codehs.com/info/curriculum/apcsp, Unit 3 on code.org: https://studio.code.org/courses/csp-201, Function Exercises at https://codepen.io/collection/noBqOm/

**Building Apps**
Students will be able design and create a program to solve problems. Students will be able to write event-driven programs. Students will be able to use variables, Boolean logic, functions with return values and lists. Students will use either python or javascript and will collaborate with a small team of students to create their program. While creating their programs, students will address the prompts which will appear on the AP Performance Task in order to practice and prepare for the AP assessment.
Resources: Unit 5 on code.org: https://studio.code.org/courses/csp-201.

**The Internet**
Students will be able to explain the abstractions in the Internet and how the Internet functions and to explain why protocols are necessary to overcome the underlying unreliability of the Internet. Students will research and present on an issue facing society that requires an understanding of how the Internet works.
**Resources:** https://ed.ted.com/lessons/what-is-the-world-wide-web-twila-camp, Unit 1 on code.org: https://studio.code.org/courses/csp-2018, *Blown to Bits* Chapter 5: http://www.bitsbook.com/excerpts/chapter5/,

**Data, Encryption and Cryptography**
Students will be able to identify existing cybersecurity concerns and potential options to address these issues with the Internet and the systems built on it.
**Resources:** NOVA Cybersecurity Lab: http://www.pbs.org/wgbh/nova/labs/lab/cyber/, CMU Cybersecurity competition: https://picoctf.com/, HS Capture the Flag: https://hsctf.com/.

## Each Unit will interweave the six Computer Science Principles Computational Thinking Practices:

P1: Connecting Computing
P2: Creating Computational Artifacts
P3: Abstracting
P4: Analyzing Problems and Artifacts
P5: Communicating (both orally and written)
P6: Collaborating

## Along with the seven Computer Science Principles Big Ideas:

Big Idea 1: Creativity
Computing is a creative activity. In this course, you will use the tools and techniques of computer science to create interesting and relevant digital artifacts (e.g., a video, animation, infographic, audio recording or program) with characteristics that are enhanced by computation.

Big Idea 2: Abstraction
Abstraction is a central problem-solving technique in computer science. In this course, you'll use abstraction to model the world and communicate with people and machines.

Big Idea 3: Data and Information
Data and information facilitate the creation of knowledge. Managing and interpreting an overwhelming amount of raw data is part of the foundation of our information society and technology. In this course, you will work with data to better understand the many ways in which data is transformed into information and knowledge.

Big Idea 4: Algorithms
Algorithms are used to develop and express solutions to computational problems. They are fundamental to even the most basic everyday task. In this course, you will work with algorithms in many ways: You will develop and express original algorithms, implement algorithms in a language, and analyze algorithms analytically and empirically.

Big Idea 5: Programming
Programming enables problem solving, human expression, and creation of knowledge. It results in the creation of software, and it facilitates the creation of computational artifacts, including music, images, and visualizations. In this course, you'll learn the fundamental concepts of programming that can be applied across a variety of projects and languages. You will create programs, translating human intention into computational artifacts.

Big Idea 6: The Internet
The Internet and systems built on it have a profound impact on society. It pervades modern computing. In this course, you will: gain insight into how the Internet operates; study characteristics of the Internet and systems built on it; and analyze important concerns, such as cybersecurity.

Big Idea 7: Global Impact
Computation has changed the way people think, work, live, and play. In this course, you'll become familiar with many ways in which computing enables innovation. You will analyze the potential benefits and harmful effects of computing in a number of contexts.

## What to do when you finish class work early

If you have successfully completed your class work, check it over. After you have finished your work and checked it, you have three options:

1. *CodingBat* problems in java or python: http://codingbat.com
2. *w3schools* tutorials in javascript, HTML and CSS: https://www.w3schools.com/
3. *Raspberry Pi challenges* which include both python programming and implementing circuits.

## Article of the Week

At the beginning of most weeks Global students will receive an article to read. Some weeks the article will be assigned in AP CSP, and sometimes it will be assigned in another Global class. For each Article of the Week (AoW) students will:

1. Read and annotate the article
2. Respond to reading for meaning statements with information from the text.
3. Write a reflection.

AoW assignments are given so that students:

- Have the background knowledge to think critically about issues in computer science.
- Deepen their understanding of the world and the role that computer science plays in it.
- Read complex informational texts and responded to them in writing.
- Use textual evidence effectively to prove or disprove text-based statements.

## Student Acknowledgment Form

I, ________________________________________, have received, read, and

(print student name)

accept the expectations, procedures, and policies for Mrs. Bailey's class.

Student signature: ___________________________________________

## Parent/Guardian Acknowledgement Form

I, ________________________________________, have received, read, and

(print parent/guardian name)

accept the expectations, procedures, and policies for Mrs. Bailey's class

with my child _____________________________________.

(print student's name here)

Parent/Guardian Signature: ___________________________________________