Summit Public Schools
Summit, New Jersey
Grade Level / Content Area: Mathematics
Length of Course: 1 Academic Year
Curriculum: AP Computer Science Principles
Updated: Spring 2019

Developed By
Brian Weinfeld

Course Description: **AP Computer Science Principles is a full year course designed for students who have successfully passed the AP Computer Science A exam. In this course, students will continue their studies in the Java programming language, adding to their already extensive knowledge. In addition, they will study computer architecture and gain an understanding of the fundamental design choices in the creation of computers and programming languages. Finally, they will learn about the underlying structure of the Internet and the systems that it was built on.**

Students will also create and prepare two reports for the CollegeBoard on independently researched and developed topics. Upon completion of this course, students will be prepared to pass the associated AP examination.

Anticipated Timetable for AP Computer Science Principles

Quarter 1

Unit 1 – Data Structures

Topic	Time Frame
Introduction to Node and Singly Linked Lists	2
Exploration of Doubly Linked Lists	2
Project #1	3
Exploration of Stacks and Queues	2
Project #2	3
Exploration of Binary Search Trees	3
Lesson on Hashes	1
Comparison of Data Structures and their runtimes/memory allocations	2
Test	1
Total	19

Unit 2 – Sorts

Topic	Time Frame
Discussion of Big O	1
Big O Algorithm Practice	1
Review of Sorts	2
Bucket Sort	1
Radix Sort	1
Quick Sort	1
Tim Sort	3
Sort Comparison / Pros and Cons	2
Test	1
Total	13

Unit 3 – Input/Output

Topic	Time Frame
Scanner class introduction	1
Scanner class utilization	1
Try/Catch Introduction	1
Finally block	1
Catching Exception and Creating Custom Exceptions	3
Project #1	3
Using the File Class	1
Using the BufferedReader/BufferedWriter class	2
Project #2	3
Test	1
Total	17

Unit 4 – Regular Expressions (Regex)

Topic	Time Frame
Introduction to Regular Expressions / History of Regex	1
Basic Regular Expression Utilization	4
Regex Algorithm Practice	1
Advanced Regex Techniques (Substitution, Back References)	4
Advanced Regex Techniques Practice	1
Project #1	3
Review	1
Test	1
Total	16

Quarter 2

Unit 5 – Create Performance Task

Topic	Time Frame
Create Performance Task	12
Total	12

Unit 6 – Switches/Gates/Machines

Topic	Time Frame
6 basic gates and gate logic	2
Building integrated circuits	1
Exploring integrated circuits	1
Project #1	3
SR Switch/Latch and Flip Flop	1
D Switch/Latch and Flip Flop	1
JK Switch/Latch and Flip Flop	1

Exploring Switches/Latches	1
Project #2	4
Test	1
Total	16

Unit 7 – Compression and Encryption

Topic	Time Frame
Introduction to compression techniques	1
Exploring common compression techniques	2
Project #1	3
P vs. NP	1
Exploration of NP-complete problems	1
Introduction to encryption techniques	1
Public and Private Keys	1
Encryption and Decryption in Java	1
Project #2	4
Total	15

Quarter 3

Unit 8 – Explore Performance Task

Topic	Time Frame
Explore Performance Task	8
Total	8

Unit 9 – The Internet

Topic	Time Frame
Introduction to the Internet	1
History of the Internet	3
The Systems the Internet is Built Upon	3
Exploration of DNS system	2
Project #1	3
Introduction to Cybersecurity	1
How Encryption is Used to Make the Internet Safer	3
Pros and Cons of Various Cybersecurity Methods	3
Project #2	4
Review	1
Test	1
Total	25

Quarter 4

Unit 10 – Review (15 days)

Topic	Time Frame

Discussion and outline of AP exam and expectations	1
Review of Quarter 1 and 2	1
Practice AP Questions on Quarter 1 and 2 Topics	1
Review of Quarter 3	1
Practice AP Questions on Quarter 3 Topics	1
Practice of Multiple-Choice	3
PRACTICE AP EXAM	3
Review of difficult topics/student sticking points	4
Total	15

Unit 11 – Java Applet Creation

Topic	**Time Frame**
How to create a Java Applet	5
Individual Project	19
Total	24

Unit 1: Data Structures

<table>
<tr><th colspan="2">Standard</th></tr>
<tr><td colspan="2">Big Ideas: Course Objectives / Content Statement(s)
● Students will be able to utilize Nodes to create various Data Structures
● Students will be able to create and utilize Linked Lists, Stacks, Queues, BST and Hashes
● Students will know the efficiency of the various actions each Data Structure
● Students will be able to select the best Data Structure for a given task</td></tr>
<tr><td>Essential Questions
What provocative questions will foster inquiry, understanding, and transfer of learning?</td><td>Enduring Understandings
What will students understand about the big ideas?</td></tr>
<tr><td>● What is a Data Structure?
● What priorities are used to select the best Data Structure for a task?
● What is a Node and how does it function as the fundamental building block of many Data Structures?</td><td>Students will understand that…
● There is no best Data Structure
● Selection of a Data Structure must be tailored to the need of the programmer.
● Runtime and Storage must be considered in selecting the appropriate Data Structure</td></tr>
<tr><td>Areas of Focus: Proficiencies (Cumulative Progress Indicators)</td><td>Examples, Outcomes, Assessments</td></tr>
<tr><td>Students will:</td><td rowspan="8">Instructional Focus (4 weeks):
● Hands-on programming time mixed with instructional classes on new topics or common problems/mistakes

Sample Assessments:
● Warm up/cool down quizzes to test concepts previously learned in class.
● Project: Students will write code that runs the “Fast Pass” system at a local amusement park
● Written Test: Students will answer analysis questions based on available AP questions

Instructional Strategies:
● Individual work with help from peers
● In-class programming time with assistance from teacher
● At home programming time

Technology Integration

● Computer/BlueJ IDE</td></tr>
<tr><td>Be familiar with the basic functioning of the major data structures</td></tr>
<tr><td>Be able to identify key pieces of information to identify the best data structure for the task at hand</td></tr>
<tr><td>Be able to highlight to pros and cons of each structure</td></tr>
<tr><td>Be able to utilize the structures efficiently in their code.</td></tr>
<tr><td></td></tr>
<tr><td></td></tr>
<tr><td></td></tr>
</table>

	Media Literacy Integration Students will investigate the Apollo Space Guidance Computer, which utilized a precursor to modern Data Structures

Unit 2: Sorts

<table>
<tr><th colspan="2">Standard</th></tr>
<tr><td colspan="2">Big Ideas: Course Objectives / Content Statement(s)
● Students will be able to explain how several major sorts function, along with their pros and cons, runtimes, and memory allocations.
● Students will be able to code several major sorts.
● Students will be able to select the best sort for any given job.
● Students will be able to calculate Big O for various algorithms.</td></tr>
<tr><td>Essential Questions
What provocative questions will foster inquiry, understanding, and transfer of learning?</td><td>Enduring Understandings
What will students understand about the big ideas?</td></tr>
<tr><td>● Which is more important when sorting, runtime or memory allocation?
● How is Big O calculated and why is it important?
● What does Big O tell you about a sort and what does it not tell you?</td><td>Students will understand that…
● Big O represents an upper bound on the runtime of an algorithm.
● There are numerous sorts for a variety of uses and can be implemented in a variety of ways.
● Improvements are constantly being made on sorts.</td></tr>
<tr><td>Areas of Focus: Proficiencies (Cumulative Progress Indicators)</td><td>Examples, Outcomes, Assessments</td></tr>
<tr><td>Students will:
Be able to code a variety of sorts.
Identify a sort from it’s code and implementation.
Calculate Big O based on code or on a description of an algorithm.</td><td rowspan="2">Instructional Focus (3 weeks):
● Hands-on programming time mixed with instructional classes on new topics or common problems/mistakes

Sample Assessments:
● 5-minute quizzes to test concepts previously learned in class.
● Project: Students will code a variety of sorts utilizing their knowledge of Data Structures
● Written Test: Students will answer analysis questions based on available AP questions

Instructional Strategies:
● Individual work with help from peers
● In-class programming time with assistance from teacher
● At home programming time

Technology Integration
Computer/BlueJ IDE

Global Perspectives

We will examine which Sorts are built into</td></tr>
</table>

	the various modern programming languages and why they are the Sort of choice for that language.

Unit 3: Input/Output

<table>
<tr><th colspan="2">Standard</th></tr>
<tr><td colspan="2">Big Ideas: Course Objectives / Content Statement(s)
Students will know be able to read from the console and read and write from files.
Students will know be able to manipulate files and folders on their computer.
Students will know be able to utilize try/catch/finally blocks to implements their code
Students will know be able to create custom Exceptions</td></tr>
<tr><td>Essential Questions
What provocative questions will foster inquiry, understanding, and transfer of learning?</td><td>Enduring Understandings
What will students understand about the big ideas?</td></tr>
<tr><td>What are the various classes that implement I/O and what is each individual class responsible for?
What is the purpose of try/catch/finally blocks and what do their associated key words do?
What is the purpose of a custom Exception?</td><td>Students will understand that…
Try/catch/finally blocks prevent runtime exceptions from crashing their program.
They are responsible for catching their Exceptions and solving the resulting exception.
They must have the appropriate permissions to access files on their computers.</td></tr>
<tr><td>Areas of Focus: Proficiencies (Cumulative Progress Indicators)</td><td>Examples, Outcomes, Assessments</td></tr>
<tr><td>Students will:
Be able to create programs that interact with files, folders and the console to allow for interactivity between the user and their program.
Be able to anticipate, catch, and solve resulting exceptions from their code.</td><td>Instructional Focus (4 weeks):
Hands-on programming time mixed with instructional classes on new topics or common problems/mistakes
In-Class single day programming assignments
Sample Assessments:
5-minute quizzes to test concepts previously learned in class.
Project: Students will modify an earlier project to allow for the reading/writing of username and password combinations to save information about each user.
Written Test: Students will answer analysis questions based on available</td></tr>
</table>

	AP questions Instructional Strategies: ● Group work/individual work ● In-class programming time with assistance from teacher ● At home programming time Technology Integration ● Computer/BlueJ IDE Global Perspectives We will examine several additional classes that allow for similar functionality as the ones we have learned. We will discuss and research why several seemingly overlapping classes have been written for Java.

Unit 4: Regular Expressions

<table>
<tr><th colspan="2">Standard</th></tr>
<tr><td colspan="2">Big Ideas: Course Objectives / Content Statement(s)
● Students will understand the importance of regular expressions, how and why they are used and why they were developed.
● Students will be able to utilize regular expressions in processing large text files.</td></tr>
<tr><td>Essential Questions
What provocative questions will foster inquiry, understanding, and transfer of learning?</td><td>Enduring Understandings
What will students understand about the big ideas?</td></tr>
<tr><td>● Why are regular expressions important and why are they considered difficult?
● How can regular expression use be combined with I/O to create and edit files?</td><td>Students will understand that…
● Regular expressions follow a precise set of rules that must be accounted for in creating and processing text files.</td></tr>
<tr><td>Areas of Focus: Proficiencies (Cumulative Progress Indicators)</td><td>Examples, Outcomes, Assessments</td></tr>
<tr><td>Students will:</td><td rowspan="6">Instructional Focus (3 weeks):
● Hands-on programming time mixed with instructional classes on new topics or common problems/mistakes
● In-Class single day programming assignments

Sample Assessments:
● 5-minute quizzes to test concepts previously learned in class.
● Project: Students will process a file on sports players to calculate various statistics on them.
● Written Test: Students will answer analysis questions based on available AP questions

Instructional Strategies:
● Group work/individual work
● In-class programming time with assistance from teacher
● At home programming time

Technology Integration
● Computer/BlueJ IDE

Global Perspectives

We will examine the creation and rise of regular expressions and why they are viewed</td></tr>
<tr><td>Students will be able to code regular expressions.</td></tr>
<tr><td>Students will be able to parse text files for data.</td></tr>
<tr><td>Be able to create new files and transfer data to them.</td></tr>
<tr><td></td></tr>
<tr><td></td></tr>
<tr><td></td></tr>
</table>

	as a necessary evil.

Unit 5: Create Performance Task

Standard	
Big Ideas: *Course Objectives / Content Statement(s)* ● **In this CollegeBoard required unit, students will research a topic of their choosing, create a program based on this research, and write an accompanying essay answering a variety of questions supplied by the CollegeBoard. They will then submit this project for grading.**	
Essential Questions *What provocative questions will foster inquiry, understanding, and transfer of learning?*	**Enduring Understandings** *What will students understand about the big ideas?*
Areas of Focus: Proficiencies (Cumulative Progress Indicators)	**Examples, Outcomes, Assessments**
	Instructional Focus (2 week): ● Students will work on their individual project. Technology Integration ● Computer/BlueJ IDE

Unit 6: Switches/Gates/Machines

<table>
<tr><th colspan="2">Standard</th></tr>
<tr><td colspan="2">Big Ideas: Course Objectives / Content Statement(s)
● Students will know the 6 different types of gates.
● Students will be able to turn equations into a system of gates and vice versa
● Students will be able to build gates to perform a variety of functions
● Students will be able to go basic binary math.</td></tr>
<tr><td>Essential Questions
What provocative questions will foster inquiry, understanding, and transfer of learning?</td><td>Enduring Understandings
What will students understand about the big ideas?</td></tr>
<tr><td>● How is a gate essential to our understanding of computers?
● What are the different types of gates and what are their functions?
● What is the relationship between decimal math and binary math?</td><td>Students will understand that…
● Gates are the basic building block of all computers.
● Each type of gate performs an essential function.
● Binary math has the same basic rules as decimal math.</td></tr>
<tr><td>Areas of Focus: Proficiencies (Cumulative Progress Indicators)</td><td>Examples, Outcomes, Assessments</td></tr>
<tr><td>Students will:
Know how to build systems of gates
Perform basic binary math operations.</td><td>Instructional Focus (3 weeks):
● Hands-on wiring on a gate simulator mixed with instructional classes on new topics or common problems/mistakes

Sample Assessments:
● 5-minute quizzes to test concepts previously learned in class.
● Project: Students will work in groups to create a system of gates that power a standard sports scoreboard display.
● Written Test: Students will answer analysis questions based on available AP questions

Instructional Strategies:
● Individual work
● In-class programming time with assistance from teacher
● At home programming time

Technology Integration
● Internet

Media Literacy Integration
We will examine the gate set up for a standard 8-digit calculator and explore how machines that require millions of gates are</td></tr>
</table>

	created.

Unit 7: Compression and Encryption

Standard

Big Ideas: *Course Objectives / Content Statement(s)*

- **Students will examine compression algorithms both in purpose and execution.**
- **Students will be able to identify the key difference between P and NP problems and understand the core importance of P=NP.**
- **Students will be able to identify common NP-complete problems.**
- **Students will be able to write basic encryption/decryption algorithms**

Essential Questions *What provocative questions will foster inquiry, understanding, and transfer of learning?*	**Enduring Understandings** *What will students understand about the big ideas?*
● If data is all 0s and 1s, how can a computer compress this information? ● How is data typically encrypted? How can we confirm that data came from a specific person? ● What are the uses of public/private keys? ● How is P=NP important to the discussion of Encryption?	Students will understand that… ● Compression algorithms require a method to refer back to previous pieces of data and a way of finding matching data. ● Encryption utilizes problems that are slow for computer to solve but quick to verify. ● P=NP is an important debate in computer science and will change the face of computing as we know it once resolved.
Areas of Focus: Proficiencies (Cumulative Progress Indicators)	**Examples, Outcomes, Assessments**
Students will: Create their own compression and encryption algorithms.	Instructional Focus (3 weeks): ● Hands-on programming time mixed with instructional classes on new topics or common problems/mistakes Instructional Strategies: ● Individual work ● Project: Students will work in groups to utilize compression algorithms in a contest to see who can most significantly compress a file. ● Written Test: Students will answer analysis questions based on available AP questions Technology Integration ● Computer/BlueJ IDE Global Perspectives Students will research our growing demand for faster computers and additional bandwidth. They will explore and present to the class several of the ideas that are

	currently being utilized to address this major issue.

Unit 8: Explore Performance Task

Standard
Big Ideas: *Course Objectives / Content Statement(s)* ● **In this CollegeBoard required unit, students will research a topic of their**

choosing that has global repercussions. They will explore this topic and how it relates to the creation and storage of data. They will create artifacts and answer several questions supplied by the CollegeBoard. They will submit this project for grading to the CollegeBoard.	
Essential Questions *What provocative questions will foster inquiry, understanding, and transfer of learning?*	**Enduring Understandings** *What will students understand about the big ideas?*
Areas of Focus: Proficiencies (Cumulative Progress Indicators)	**Examples, Outcomes, Assessments**
	Instructional Focus (2 week): ● Students will work on their individual project. Technology Integration ● Computer

Unit 9: The Internet

Standard
Big Ideas: *Course Objectives / Content Statement(s)*

<table>
<tr><td colspan="2">

- **Students will understand that the Internet is a network of autonomous systems.**
- **Students will learn of domain name syntax, IP addresses and routing.**
- **Students will understand the importance of Cybersecurity.**

</td></tr>
<tr><th>Essential Questions
What provocative questions will foster inquiry, understanding, and transfer of learning?</th><th>Enduring Understandings
What will students understand about the big ideas?</th></tr>
<tr><td>

- How does the Internet function? Who runs the Internet?
- What is the core structure of the Internet?
- What are the Internet's safety protocols?
- What is Cybersecurity?

</td><td>Students will understand that…

- The Internet is built on a series of redundant hierarchical systems.
- Cybersecurity is as much an art as a science and still leaves gaps in security.

</td></tr>
<tr><th>Areas of Focus: Proficiencies (Cumulative Progress Indicators)</th><th>Examples, Outcomes, Assessments</th></tr>
<tr><td>Students will:
Be able to describe and explain the core system upon which the Internet is built.
Identify key redundancies in the Internet's core structure
Identify new creations in CyberSecurity and it's importance as it relates to P vs. NP.</td><td>Instructional Focus (5 weeks):

- Hands-on programming time mixed with instructional classes on new topics or common problems/mistakes

Instructional Strategies:

- Individual work
- Project: Students will work in groups to research common security systems and identify their pros and limitations.
- Written Test: Students will answer analysis questions based on available AP questions

Technology Integration

- Computer with Internet Access

Global Perspectives
Students will research the growth of the Internet and how its redundant systems aided in its growth.</td></tr>
</table>

Unit 10: Review

<table>
<tr><th colspan="2">Standard</th></tr>
<tr><td colspan="2">Big Ideas: Course Objectives / Content Statement(s)

- **Students will be prepared to take the AP Computer Science Principles Exam**

</td></tr>
<tr><th>Essential Questions</th><th>Enduring Understandings</th></tr>
</table>

What provocative questions will foster inquiry, understanding, and transfer of learning?	*What will students understand about the big ideas?*
● What is left for me to learn or review to prepare for the exam?	Students will understand that… ●
Areas of Focus: Proficiencies (Cumulative Progress Indicators)	**Examples, Outcomes, Assessments**
Students will:	Instructional Focus (3 weeks): ● Hands-on programming time mixed with instructional classes on new topics or common problems/mistakes Sample Assessments: ● 5-minute quizzes to test concepts previously learned in class. ● Written Test: Students will be given a full AP exam for practice. Instructional Strategies: ● Individual work ● In-class programming time with assistance from teacher ● At home programming time Technology Integration ● None required

Unit 11: Final Project

Standard
Big Ideas: *Course Objectives / Content Statement(s)* ● **Students will develop and design their own final project. They will examine and explore this topic with assistance from myself.**

Essential Questions *What provocative questions will foster inquiry, understanding, and transfer of learning?*	**Enduring Understandings** *What will students understand about the big ideas?*
● What topic that we learned this year would I like to learn more about?	Students will understand that… ●
Areas of Focus: Proficiencies (Cumulative Progress Indicators)	**Examples, Outcomes, Assessments**
Students will:	Instructional Focus (~4 weeks): ● Individual/group time to focus on a topic of their choosing. Sample Assessments: ● Presentation of material to class. Instructional Strategies: ● Individual/group work ● Specialized time with teacher Technology Integration ● Any previously listed technology or an additional new technology with my approval.