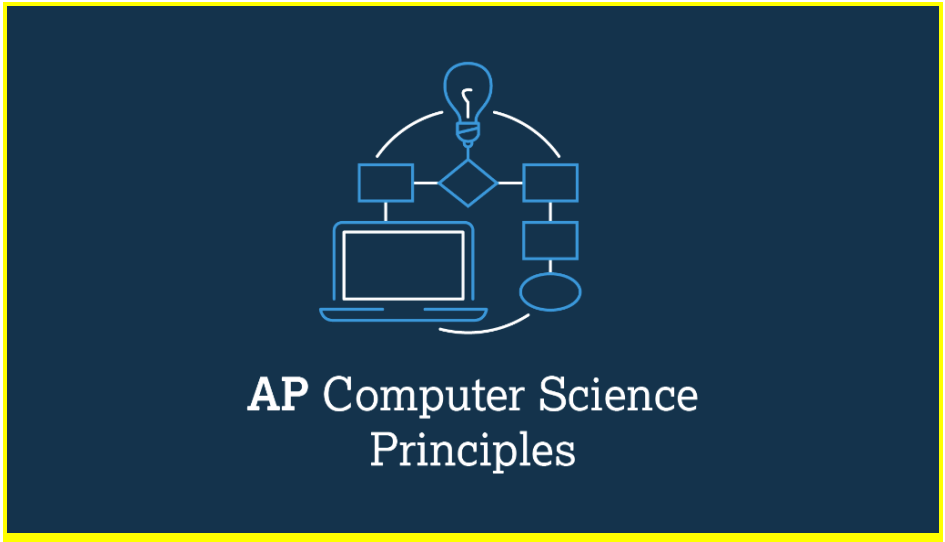
# AP COMPUTER SCIENCE PRINCIPLES CURRICULUM





**Grade Level(s): 10-12**

**Curriculum Author(s):** Raymond Robillard

**Course Description:** The AP Computer Science Principles course opens up the world of Computer Science to all students by providing a CS course with a greater focus on fundamental concepts in CS without getting bogged down in the exact details of syntax for a specific programming language. This course covers both universal concepts in programming but also focuses on concepts such as: digital copyright issues, closing the technology gap created by economic inequity, how the internet functions, ethical issues in CS, data abstraction, and algorithms. The course has a heavy emphasis on collaboration. The AP Computer Science Principles course curriculum is compatible with many CS1 courses in colleges and universities. Students will be able to obtain college credits by successfully meeting a benchmark score on the Advanced Placement Exam. Block coding is utilized in the course through the online platform at code.org.

| Year At A Glance | | | |
|---|---|---|---|
| **Unit Title** | **Overarching Essential Question** | **Overarching Enduring Understanding** | **Vision of A Learner "I Can" Statements** |
| Digital Information | • How can we use 1s and 0s to represent something complex like a video of the marching band playing a song?<br>• How are vastly different kinds of data, physical phenomena, and mathematical concepts represented on a computer?<br>• How accurately can human experience and perception be captured or reflected in digital information? | • The way that the computer represents data is different from the way that the data are interpreted and displayed for the user. Programs are used to translate data into a representation that is more easily understood by people.<br>• Abstraction is the process of reducing complexity by focusing on the main idea. By hiding details irrelevant to the question at hand and bringing together related and useful details, abstraction reduces complexity and allows one to focus on the idea. | TCC2(9-12), TCC4(9-12), TI3(9-12) |
| The Internet | • What is the internet? How is it built? How does it function?<br>• What aspects of the internet's design and development have helped it scale and flourish? | • Computer systems and networks facilitate how data is transferred.<br>• The Internet is a computer network consisting of interconnected networks that use standardized, open (nonproprietary) communication protocols. | TCC4(9-12), TI3(9-12) |
| Intro to App Design | • How are programs developed to help people, organizations, or society solve problems?<br>• How are programs used for creative expression, to satisfy personal curiosity, or to create new knowledge? | • Developers create and innovate using an iterative design process that is user-focused, that incorporates implementation/feedback cycles, and that leaves ample room for experimentation and risk-taking.<br>• The way statements are sequenced and combined in a program determines the computed result. Programs incorporate iteration and selection constructs to represent repetition and make decisions to handle varied input values. | TCC1(9-12), TCC3(9-12), TCC4(9-12) |

| | | | |
|---|---|---|---|
| Variables, Conditionals, and Functions | • How can we store data in a program to solve problems?<br>• What is an abstraction?<br>• How does abstraction help us in writing programs, creating computational artifacts, and solving problems? | • To find specific solutions to generalizable problems, programmers represent and organize data in multiple ways.<br>• Programmers break down problems into smaller and more manageable pieces. By creating procedures and leveraging parameters, programmers generalize processes that can be reused. Procedures allow programmers to draw upon existing code that has already been tested, allowing programmers to write programs more quickly and with more confidence. | TI3(9-12), AA1(9-12) |
| Lists, Loops, and Traversals | • How does abstraction make the development of computer programs possible?<br>• What are some advantages to using data structures such as lists? | • Developing a data abstraction, like a list, to implement in a program can result in a program that is easier to develop and maintain.<br>• Iteration statements change the sequential flow of control by repeating a set of statements zero or more times, until a stopping condition is met.<br>• Traversing a list can be a complete traversal, where all elements in the list are accessed, or a partial traversal, where only a portion of elements are accessed. | TCC3(9-12), TI3(9-12), AA1(9-12) |
| Algorithms | • What types of problems can be solved more easily with a computer, and what types can be solved more easily without a computer? Why?<br>• How are algorithms implemented and executed on computers and computational devices? | • Every algorithm can be constructed using combinations of sequencing, selection, and iteration.<br>• There exist problems that the computer cannot solve, and even when the computer can solve a problem, it may not be able to do so in a reasonable amount of time. | TCC1(9-12), TCC3(9-12), TCC4(9-12), AA1(9-12) |
| Parameters, Return, and Libraries | • What components of an algorithmic process are needed | • A procedural abstraction may extract shared features to generalize | TI3(9-12), AA1(9-12) |

| | | | |
|---|---|---|---|
| | for it to produce a meaningful result?<br>• Why is code reuse important in the field of Computer Science? | functionality instead of duplicating code. This allows for program code reuse, which helps manage complexity.<br>• Using parameters allows procedures to be generalized, enabling the procedures to be reused with a range of input values or arguments. | |
| Data | • How can computation be employed to help people process data and information to gain insight and knowledge?<br>• When is it more appropriate to use a computer to analyze data than to complete the analysis by hand?<br>• Where and how does human bias affect the collection, processing and interpretation of data? | • Data provide opportunities for identifying trends, making connections, and addressing problems.<br>• While computing innovations are typically designed to achieve a specific purpose, they may have unintended consequences.<br>• Responsible programmers try to consider the unintended ways their computing innovations can be used and the potential beneficial and harmful effects of these new uses. | TCC1(9-12), TCC2(9-12) |
| Cybersecurity and Global Impacts | • What app or computer software do you use most often and would have a hard time going without? How does this software solve a problem for you or benefit you?<br>• Are innovators responsible for the harmful effects of their computing innovations, even if those effects were unintentional? Why or why not?<br>• How is cyber security impacting the ever-increasing number of internet users? | • The use of computing innovations may involve risks to your personal safety and identity.<br>• Responsible programmers try to consider the unintended ways their computing innovations can be used and the potential beneficial and harmful effects of these new uses. | TCC1(9-12), TCC2(9-12), TCC4(9-12), AA4(9-12) |

# Unit 1 - Digital Information

## Desired Results - Goals, Transfer, Meaning, Acquisition

**Established Goals:**
2-DA-07 - Represent data using multiple encoding schemes.
2-IC-20 - Compare tradeoffs associated with computing technologies that affect people's everyday activities and career options.
3A-AP-21 - Evaluate and refine computational artifacts to make them more usable and accessible.
3A-CS-02 - Compare levels of abstraction and interactions between application software, system software and hardware layers.
3A-DA-09 - Translate between different bit representations of real-world phenomena, such as characters, numbers, and images.
3A-DA-10 - Evaluate the tradeoffs in how data elements are organized and where data is stored.
3A-IC-24 - Evaluate the ways computing impacts personal, ethical, social, economic, and cultural practices.
3A-IC-28 - Explain the beneficial and harmful effects that intellectual property laws can have on innovation.
3B-IC-27 - Predict how computational innovations that have revolutionized aspects of our culture might evolve.

**Vision of A Learner Attributes:**

TCC2(9-12): I can evaluate evidence from multiple perspectives, and recognize their limitations and implications, in order to justify new conclusions.
TCC4(9-12): I can integrate my learning to adapt to experiences in the classroom, career and life.
TI3(9-12): I can formulate and investigate probing questions to further my learning.

**Understandings:** Students will understand that...

- The way that the computer represents data is different from the way that the data are interpreted and displayed for the user. Programs are used to translate data into a representation that is more easily understood by people.
- Abstraction is the process of reducing complexity by focusing on the main idea. By hiding details irrelevant to the question at hand and bringing together related and useful details, abstraction reduces complexity and allows one to focus on the idea.

**Essential Questions:**

- How can we use 1s and 0s to represent something complex like a video of the marching band playing a song?
- How are vastly different kinds of data, physical phenomena, and mathematical concepts represented on a computer?
- How accurately can human experience and perception be captured or reflected in digital information?

**Students will know...**

- In programming languages, the fixed number of bits used to represent integers and real numbers limits the range and mathematical operations on these values; this limitation can result in round-off and overflow errors. Some real numbers are

**Students will be able to...**

- Explore how computers store complex information like numbers, text, images and sound and debate the impacts of digitizing information.
- Explain the consequences of using bits to represent data.

| | |
|---|---|
| represented as approximations in computer storage.<br>• Analog data can be closely approximated digitally using a sampling technique, which means measuring values of the analog signal at regular intervals called samples. The samples are measured to figure out the exact bits required to store each sample.<br>• Data compression can reduce the size (number of bits) of transmitted or stored data.<br>• Lossy data compression algorithms can significantly reduce the number of bits stored or transmitted but only allow reconstruction of an approximation of the original data.<br>• Material created on a computer is the intellectual property of the creator or an organization.<br>• The use of material created by someone other than you should always be cited. | • Compare data compression algorithms to determine which is best in a particular context.<br>• Explain how the use of computing could raise legal and ethical concerns. |
| **Key Vocabulary:** Binary, Bit, Byte, Overflow Error, Round-off Error, Analog Data, Digital Data, Sampling, Lossless Compression, Lossy Compression, Intellectual Property, Creative Commons | |
| **Assessment Evidence** | |
| **Performance Tasks:**<br>Project Digital Information Dilemmas Part 1:<br>Students begin a two day project where they will demonstrate their understanding of key issues surrounding digital information. This project is designed to be used in tandem with the Unit 1 Assessment to evaluate student progress in Unit 1 content. You may find after this assessment that students are confused about how some things are represented digitally. This is a good opportunity to go back and review key takeaways from previous lessons before continuing on to the Unit 1 Assessment. | **Other Evidence:**<br>At the conclusion of this unit, students will complete a multiple choice assessment of AP-like problems to both assess their understanding of core concepts and to prepare them for the AP Exam they will take in May. |
| **Learning Plan** | |
| This unit explores the technical challenges and questions that arise from the need to represent digital information in computers. Students learn how complex information like numbers, text, images, and sound are represented in text, how compression works, and the broader social impacts of digitizing the world's information. | |
| **Teacher Resources:** | |

Code.org AP CSP Online Materials - https://studio.code.org/courses/csp-2022?section_id=3618538
Supplemental Resources:
http://chortle.ccsu.edu/Java5/index.html#07
Java Runtime Environment
Java Development Kit

# Unit 2 - The Internet

## Desired Results - Goals, Transfer, Meaning, Acquisition

**Established Goals:**
2-NI-04 - Model the role of protocols in transmitting data across networks and the Internet.
3A-NI-04 - Evaluate the scalability and reliability of networks, by describing the relationship between routers, switches, servers, topology, and addressing.
3A-IC-24 - Evaluate the ways computing impacts personal, ethical, social, economic, and cultural practices.
3A-IC-28 - Explain the beneficial and harmful effects that intellectual property laws can have on innovation.
3A-IC-30 - Evaluate the social and economic implications of privacy in the context of safety, law, or ethics.
3B-IC-26 - Evaluate the impact of equity, access, and influence on the distribution of computing resources in a global society.
3B-IC-28 - Debate laws and regulations that impact the development and use of software.
3B-NI-03 - Describe the issues that impact network functionality (e.g., bandwidth, load, delay, topology).

**Vision of A Learner Attributes:**
TCC4(9-12): I can integrate my learning to adapt to experiences in the classroom, career and life.
TI3(9-12): I can formulate and investigate probing questions to further my learning.

**Understandings:** Students will understand that...
- Computer systems and networks facilitate how data is transferred.
- The Internet is a computer network consisting of interconnected networks that use standardized, open (nonproprietary) communication protocols.

**Essential Questions:**
- What is the internet? How is it built? How does it function?
- What aspects of the internet's design and development have helped it scale and flourish?

**Students will know…**
- A computing device is a physical artifact that can run a program. Some examples include computers, tablets, servers, routers, and smart sensors.
- Routing is the process of finding a path from sender to receiver.
- A protocol is an agreed-upon set of rules that specify the behavior of a system.
- The Internet has been engineered to be fault-tolerant, with abstractions for routing and transmitting data.
- When a system can support failures and still continue to function, it is called fault-tolerant. This is important because

**Students will be able to…**
- Learn about how the Internet works and discuss its impacts on politics, culture, and the economy.
- Explain how computing devices work together in a network.
- Explain how the Internet works.
- Describe the differences between the Internet and the World Wide Web.
- Explain how computing resources can be protected and can be misused.

| | |
|---|---|
| elements of complex systems fail at unexpected times, often in groups, and fault tolerance allows users to continue to use the network.<br>• Information is passed through the Internet as a data stream. Data streams contain chunks of data, which are encapsulated in packets.<br>• HTTP is a protocol used by the World Wide Web. | |

**Key Vocabulary:** Computing Device, Computing System, Computing Network, Path, Bandwidth, Protocol, IP Address, Internet Protocol(IP), Router, Packet, Redundancy, Fault Tolerant, HTTP, Internet, World Wide Web, Digital Divide

**Assessment Evidence**

| | |
|---|---|
| **Performance Tasks:**<br>Project Internet Dilemmas:<br>In this project, students explore a relevant Internet dilemma: Net Neutrality, Internet Censorship, or the Digital Divide. Students apply their knowledge of how the Internet works to address the core question related to their chosen dilemma. This project addresses the "so what" question - why is it important to learn about how the Internet works? | **Other Evidence:**<br>At the conclusion of this unit, students will complete a multiple choice assessment of AP-like problems to both assess their understanding of core concepts and to prepare them for the AP Exam they will take in May. |

**Learning Plan**

This unit reveals how the Internet was designed to connect billions of devices and people to one another. Students learn how the different protocols of the Internet work and actually build them themselves using the Internet Simulator. They then consider the impacts the Internet has had, both good and bad, on modern life.

**Teacher Resources:**
Code.org AP CSP Online Materials - https://studio.code.org/courses/csp-2022?section_id=3618538
Supplemental Resources:
http://chortle.ccsu.edu/Java5/index.html#07
Java Runtime Environment
Java Development Kit

# Unit 3 - Intro to App Design

## Desired Results - Goals, Transfer, Meaning, Acquisition

**Established Goals:**
2-AP-17 - Systematically test and refine programs using a range of test cases.
3A-AP-13 - Create prototypes that use algorithms to solve computational problems by leveraging prior student knowledge and personal interests.
3A-AP-16 - Design and iteratively develop computational artifacts for practical intent, personal expression, or to address a societal issue by using events to initiate instructions.
3A-AP-19 - Systematically design and develop programs for broad audiences by incorporating feedback from users.
3A-AP-21 - Evaluate and refine computational artifacts to make them more usable and accessible.
3A-AP-22 - Design and develop computational artifacts working in team roles using collaborative tools.
3A-AP-23 - Document design decisions using text, graphics, presentations, and/or demonstrations in the development of complex programs.
3A-CS-03 - Develop guidelines that convey systematic troubleshooting strategies that others can use to identify and fix errors.

**Vision of A Learner Attributes:**
TCC1(9-12): I can ask purposeful, insightful questions to find a variety of innovative solutions.
TCC3(9-12): I can integrate relevant information to produce multiple valid solutions.
TCC4(9-12): I can integrate my learning to adapt to experiences in the classroom, career and life.

**Understandings:** Students will understand that...
- Developers create and innovate using an iterative design process that is user-focused, that incorporates implementation/feedback cycles, and that leaves ample room for experimentation and risk-taking.
- The way statements are sequenced and combined in a program determines the computed result. Programs incorporate iteration and selection constructs to represent repetition and make decisions to handle varied input values.

**Essential Questions:**
- How are programs developed to help people, organizations, or society solve problems?
- How are programs used for creative expression, to satisfy personal curiosity, or to create new knowledge?

**Students will know…**
- Program input is data sent to a computer for processing by a program. Input can come in a variety of forms, such as tactile, audio, visual, or text.
- Program output is any data sent from a program to a device. Program output can come in a variety of forms, such as tactile, audio, visual, or text.

**Students will be able to…**
- Design their first app while learning both fundamental programming concepts and collaborative software development processes.
- Describe the purpose of a computing innovation.
- Identify input(s) to a program.
- Identify output(s) produced by a program.

| | |
|---|---|
| • Sequential statements execute in the order they appear in the code segment.<br>• A program is a collection of program statements that performs a specific task when run by a computer. A program is often referred to as software.<br>• An event is associated with an action and supplies input data to a program.<br>• In event-driven programming, program statements are executed when triggered rather than through the sequential flow of control.<br>• Program documentation is a written description of the function of a code segment, event, procedure, or program and how it was developed. | • Represent a step-by-step algorithmic process using sequential code statements.<br>• Explain how a program or code segment functions. |
| **Key Vocabulary:** User Interface, Input, Output, Program Statement, Program, Sequential Programming, Event Driven Programming, Documentation, Comment, Pair Programming, Debugging, Development Process, Event | |
| **Assessment Evidence** | |
| **Performance Tasks:**<br>Project Designing and App:<br>This lesson kicks off a project that students will complete throughout the unit. The framing of this project is also important for how programming is presented overall. Students are encouraged to collaboratively design their projects, choose topics of personal interest, and build an app to meet the needs of other people. All of this is important as part of framing programming as a collaborative, creative, and socially situated pursuit. | **Other Evidence:**<br>At the conclusion of this unit, students will complete a multiple choice assessment of AP-like problems to both assess their understanding of core concepts and to prepare them for the AP Exam they will take in May. |
| **Learning Plan** | |
| This unit is an introduction to programming and app design with a heavy focus on important skills like debugging, pair programming, and user testing. Students learn how to design user interfaces and write event-driven programs in App Lab and then design a project that teaches their classmates about a topic of their choosing. | |
| **Teacher Resources:**<br>Code.org AP CSP Online Materials - https://studio.code.org/courses/csp-2022?section_id=3618538<br>Supplemental Resources: | |

http://chortle.ccsu.edu/Java5/index.html#07
Java Runtime Environment
Java Development Kit

# Unit 4 - Variables, Conditionals, and Functions

## Desired Results - Goals, Transfer, Meaning, Acquisition

**Established Goals:**
2-AP-10 - Use flowcharts and/or pseudocode to address complex problems as algorithms.
2-AP-11 - Create clearly named variables that represent different data types and perform operations on their values.
2-AP-12 - Design and iteratively develop programs that combine control structures, including nested loops and compound conditionals.
2-AP-19 - Document programs in order to make them easier to follow, test, and debug.
3A-AP-15 - Justify the selection of specific control structures when tradeoffs involve implementation, readability, and program performance and explain the benefits and drawbacks of choices made.
3A-AP-16 - Design and iteratively develop computational artifacts for practical intent, personal expression, or to address a societal issue by using events to initiate instructions.
3A-AP-17 - Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects.
3B-AP-14 - Construct solutions to problems using student-created components, such as procedures, modules and/or objects.
3B-AP-21 - Develop and use a series of test cases to verify that a program performs according to its design specifications.
3B-AP-23 - Evaluate key qualities of a program through a process such as a code review.

**Vision of A Learner Attributes:**
TI3(9-12): I can formulate and investigate probing questions to further my learning.
AA1(9-12): I can evaluate different approaches and justify the best pathway to success.

**Understandings:** Students will understand that...
- To find specific solutions to generalizable problems, programmers represent and organize data in multiple ways.
- Programmers break down problems into smaller and more manageable pieces. By creating procedures and leveraging parameters, programmers generalize processes that can be reused. Procedures allow programmers to draw upon existing code that has already been tested, allowing programmers to write programs more quickly and with more confidence.

**Essential Questions:**
- How can we store data in a program to solve problems?
- What is an abstraction?
- How does abstraction help us in writing programs, creating computational artifacts, and solving problems?

**Students will know…**
- A variable is an abstraction inside a program that can hold a value. Each variable has associated data storage that represents one value at a time, but that value can be a list or

**Students will be able to…**
- Expand the types of apps they can create by adding the ability to store information, make decisions, and better organize code.
- Represent a value with a variable.

| | |
|---|---|
| other collection that in turn contains multiple values.<br>• The assignment operator allows a program to change the value represented by a variable.<br>• A string is an ordered sequence of characters.<br>• An expression can consist of a value, a variable, an operator, or a procedure call that returns a value.<br>• Arithmetic operators are part of most programming languages and include addition, subtraction, multiplication, division, and modulus operators.<br>• Using meaningful variable names helps with the readability of program code and understanding of what values are represented by the variables.<br>• A Boolean value is either true or false.<br>• Conditional statements or "if-statements" affect the sequential flow of control by executing different statements based on the value of a Boolean expression.<br>• A logic error is a mistake in the algorithm or program that causes it to behave incorrectly or unexpectedly.<br>• A syntax error is a mistake in the program where the rules of the programming language are not followed.<br>• A run-time error is a mistake in the program that occurs during the execution of a program. Programming languages define their own run-time errors.<br>• An overflow error is an error that occurs when a computer attempts to handle a number that is outside of the defined range of values.<br>• A procedure is a named group of programming instructions that may have parameters and return values. | • Determine the value of a variable as a result of an assignment.<br>• Evaluate expressions that use arithmetic operators.<br>• Evaluate expressions that manipulate strings.<br>• Identify inputs and corresponding expected output or behaviors that can be used to check the correctness of an algorithm or program.<br>• Write statements to call procedures and determine the result or effect of a procedure call. |
| **Key Vocabulary:** Expression, Assignment Operator, Variable, String, Boolean Value, Comparison Operators, Logical Operator, Conditional Statement, Function, Function Call | |
| **Assessment Evidence** | |
| **Performance Tasks:**<br>Project Decision Maker App:<br>The Practice PT gives students the opportunity to design and program an app from scratch. Welcome to The Decision Maker App! Students | **Other Evidence:**<br>At the conclusion of this unit, students will complete a multiple choice assessment of AP-like problems to both assess their understanding of core concepts and to prepare them for the AP Exam they will take in |

| | |
|---|---|
| demonstrate mastery of variables, conditionals, and functions by combining these elements into a useful program designed to solve the problem of making a decision. | May. |
| **Learning Plan** | |
| This unit explores how variables, conditionals, and functions allow for the design of increasingly complex apps. Students learn how to program with these three new concepts through a sequence of collaborative activities. They then build their own decision maker app to share with friends and help them make a decision. | |
| **Teacher Resources:**<br>Code.org AP CSP Online Materials - https://studio.code.org/courses/csp-2022?section_id=3618538<br>Supplemental Resources:<br>http://chortle.ccsu.edu/Java5/index.html#07<br>Java Runtime Environment<br>Java Development Kit | |

# Unit 5 - Lists, Loops, and Traversals

## Desired Results - Goals, Transfer, Meaning, Acquisition

**Established Goals:**
3A-AP-14 - Use lists to simplify solutions, generalizing computational problems instead of repeated use of simple variables.
3A-AP-15 - Justify the selection of specific control structures when tradeoffs involve implementation, readability, and program performance and explain the benefits and drawbacks of choices made.
3A-AP-16 - Design and iteratively develop computational artifacts for practical intent, personal expression, or to address a societal issue by using events to initiate instructions.
3A-AP-22 - Design and develop computational artifacts working in team roles using collaborative tools.
3A-AP-23 - Document design decisions using text, graphics, presentations, and/or demonstrations in the development of complex programs.
3A-DA-12 - Create computational models that represent the relationships among different elements of data collected from a phenomenon or process.
3B-AP-10 - Use and adapt classic algorithms to solve computational problems.
3B-AP-23 - Evaluate key qualities of a program through a process such as a code review.

**Vision of A Learner Attributes:**
TCC3(9-12): I can integrate relevant information to produce multiple valid solutions.
TI3(9-12): I can formulate and investigate probing questions to further my learning.
AA1(9-12): I can evaluate different approaches and justify the best pathway to success.

**Understandings:** Students will understand that...
- Developing a data abstraction, like a list, to implement in a program can result in a program that is easier to develop and maintain.
- Iteration statements change the sequential flow of control by repeating a set of statements zero or more times, until a stopping condition is met.
- Traversing a list can be a complete traversal, where all elements in the list are accessed, or a partial traversal, where only a portion of elements are accessed.

**Essential Questions:**
- How does abstraction make the development of computer programs possible?
- What are some advantages to using data structures such as lists?

**Students will know…**
- A list is an ordered sequence of elements.
- List procedures are implemented in accordance with the syntax

**Students will be able to…**
- Build apps that use large amounts of information and pull in data from the web to create a wider variety of apps.

| | |
|---|---|
| rules of the programming language.<br>• The use of lists allows multiple related items to be treated as a single value.<br>• Simulations are abstractions of more complex objects or phenomena for a specific purpose.<br>• Random number generators can be used to simulate the variability that exists in the real world.<br>• Algorithms that appear similar can yield different side effects or results.<br>• Iteration statements can be used to traverse a list. | • Represent a list using a variable.<br>• Write expressions that use list indexing and list procedures and evaluate expressions that use list indexing and list procedures.<br>• Represent repeated code execution using iteration statements and determine the result or side-effect of iteration statements.<br>• Compare multiple algorithms to determine if they yield the same side effect or result. |
| **Key Vocabulary:** List, Element, Index, Iteration, Infinite Loop, Traversal, Data Abstraction | |
| **Assessment Evidence** | |
| **Performance Tasks:**<br>Project Hackathon:<br>Students will demonstrate their app design and programming skills throughout this five day project. In addition, students work with a dataset as this ensures students will be using the types of programming constructs required for the Create Performance Task, which this project is designed as a practice for. Students complete the project by individually filling out a Written Response, modeled after the Create PT. | **Other Evidence:**<br>At the conclusion of this unit, students will complete a multiple choice assessment of AP-like problems to both assess their understanding of core concepts and to prepare them for the AP Exam they will take in May. |
| **Learning Plan** | |
| This unit introduces lists, loops, and traversals, and explores the way they can be used to build apps that store and process large amounts of information. Students learn to program with the data library in App Lab and complete a 5-day hackathon project at the end of the unit where they can design a program about any topic of their choosing. | |
| **Teacher Resources:**<br>Code.org AP CSP Online Materials - https://studio.code.org/courses/csp-2022?section_id=3618538<br>Supplemental Resources:<br>http://chortle.ccsu.edu/Java5/index.html#07<br>Java Runtime Environment<br>Java Development Kit | |

# Unit 6 - Algorithms

## Desired Results - Goals, Transfer, Meaning, Acquisition

**Established Goals:**
3B-AP-10 - Use and adapt classic algorithms to solve computational problems.
3B-AP-11 - Evaluate algorithms in terms of their efficiency, correctness, and clarity.

**Vision of A Learner Attributes:**

TCC1(9-12): I can ask purposeful, insightful questions to find a variety of innovative solutions.
TCC3(9-12): I can integrate relevant information to produce multiple valid solutions.
TCC4(9-12): I can integrate my learning to adapt to experiences in the classroom, career and life.
AA1(9-12): I can evaluate different approaches and justify the best pathway to success.

**Understandings:** Students will understand that...

- Every algorithm can be constructed using combinations of sequencing, selection, and iteration.
- There exist problems that the computer cannot solve, and even when the computer can solve a problem, it may not be able to do so in a reasonable amount of time.

**Essential Questions:**

- What types of problems can be solved more easily with a computer, and what types can be solved more easily without a computer? Why?
- How are algorithms implemented and executed on computers and computational devices?

**Students will know…**

- An algorithm is a finite set of instructions that accomplish a specific task.
- Algorithms can be written in different ways and still accomplish the same tasks.
- Linear search or sequential search algorithms check each element of a list, in order, until the desired value is found or all elements in the list have been checked.
- Data must be in sorted order to use the binary search algorithm.
- Binary search is often more efficient than sequential/linear search when applied to sorted data.
- An algorithm's efficiency can be informally measured by determining the number of times a statement or group of statements executes.
- A heuristic is an approach to a problem that produces a solution

**Students will be able to…**

- Design and analyze algorithms to understand how they work and why some are considered better than others.
- Express an algorithm that uses sequencing without using a programming language.
- Represent a step-by-step algorithmic process using sequential code statements.
- Explain the existence of undecidable problems in computer science.
- Compare problem solutions that use sequential, parallel, and distributed computing.
- Determine the efficiency of sequential and parallel solutions.

| | |
|---|---|
| that is not guaranteed to be optimal but may be used when techniques that are guaranteed to always find an optimal solution are impractical.<br>• Sequential computing is a computational model in which operations are performed in order one at a time.<br>• Parallel computing is a computational model where the program is broken into multiple smaller sequential computing operations, some of which are performed simultaneously.<br>• Distributed computing is a computational model in which multiple devices are used to run a program. | |

**Key Vocabulary:** Problem, Algorithm, Sequencing, Selection, Iteration, Efficiency, Linear Search, Binary Search, Reasonable Time, Unreasonable Time, Heuristic, Decision Problem, Optimization Problem, Undecidable Problem, Sequential Computing, Parallel Computing, Speedup

**Assessment Evidence**

| | |
|---|---|
| **Performance Tasks:**<br>There is no performance task specific to this unit. | **Other Evidence:**<br>At the conclusion of this unit, students will complete a multiple choice assessment of AP-like problems to both assess their understanding of core concepts and to prepare them for the AP Exam they will take in May. |

**Learning Plan**

This unit is a quick exploration of how computer scientists design algorithms to solve problems and how they analyze the speed of different algorithms. Students learn about the concept of algorithmic efficiency through a variety of hands-on activities and learn how it's being applied in modern computing.

**Teacher Resources:**
Code.org AP CSP Online Materials - https://studio.code.org/courses/csp-2022?section_id=3618538
Supplemental Resources:
http://chortle.ccsu.edu/Java5/index.html#07
Java Runtime Environment
Java Development Kit

# Unit 7 - Parameters, Return, and Libraries

## Desired Results - Goals, Transfer, Meaning, Acquisition

**Established Goals:**
2-AP-14 - Create procedures with parameters to organize code and make it easier to reuse.
2-AP-17 - Systematically test and refine programs using a range of test cases.
3A-AP-18 - Create artifacts by using procedures within a program, combinations of data and procedures, or independent but interrelated programs.
3B-AP-14 - Construct solutions to problems using student-created components, such as procedures, modules and/or objects.
3B-AP-16 - Demonstrate code reuse by creating programming solutions using libraries and APIs.
3B-AP-23 - Evaluate key qualities of a program through a process such as a code review.

**Vision of A Learner Attributes:**
TI3(9-12): I can formulate and investigate probing questions to further my learning.
AA1(9-12): I can evaluate different approaches and justify the best pathway to success.

**Understandings:** Students will understand that...
- A procedural abstraction may extract shared features to generalize functionality instead of duplicating code. This allows for program code reuse, which helps manage complexity.
- Using parameters allows procedures to be generalized, enabling the procedures to be reused with a range of input values or arguments.

**Essential Questions:**
- What components of an algorithmic process are needed for it to produce a meaningful result?
- Why is code reuse important in the field of Computer Science?

**Students will know…**
- Parameters are input variables of a procedure. Arguments specify the values of the parameters when a procedure is called.
- Using procedural abstraction helps improve code readability.
- A software library contains procedures that may be used in creating new programs.
- Existing code segments can come from internal or external sources, such as libraries or previously written code.
- The subdivision of a computer program into separate subprograms is called modularity.

**Students will be able to…**
- Learn how to design clean and reusable code that you can share with a single classmate or the entire world.
- Write statements to call procedures and determine the result or effect of a procedure call.
- Select appropriate libraries or existing code segments to use in creating new programs.

**Key Vocabulary:** Parameter, Argument, Return, Procedural Abstraction, Library, API

## Assessment Evidence

**Performance Tasks:**
Project Make a Library:

This project caps off everything students have learned about programming as they head into the Create PT. Students are literally "building blocks" that classmates will literally have access to in order to create projects of their own. This kind of thinking requires not only a mastery of programming concepts and skills, but the ability to think more abstractly about how programs are built. Students need to identify common situations or problems that they and other programmers may encounter, and help build commands to address that problem.

This project also caps off a unit that is all about abstraction. On one level students are thinking about procedural abstraction, and simply learning how to design functions. On another level, however, they're learning to think abstractly. They're not merely designing a function, but they're thinking about the entire process of how programs are developed and common problems or situations that arise. This mental approach to thinking about why to build a function is just as important as knowing the steps to design one.

**Other Evidence:**
At the conclusion of this unit, students will complete a multiple choice assessment of AP-like problems to both assess their understanding of core concepts and to prepare them for the AP Exam they will take in May.

## Learning Plan

This unit introduces parameters, return, and libraries. Students learn how to use these concepts to build new kinds of apps as well as libraries of code that they can share with their classmates. Students end the unit by designing a library of functions around any topic of their choosing.

**Teacher Resources:**
Code.org AP CSP Online Materials - https://studio.code.org/courses/csp-2022?section_id=3618538
Supplemental Resources:
http://chortle.ccsu.edu/Java5/index.html#07
Java Runtime Environment
Java Development Kit

# Unit 8 - Data

## Desired Results - Goals, Transfer, Meaning, Acquisition

**Established Goals:**
3A-DA-10 - Evaluate the tradeoffs in how data elements are organized and where data is stored.
3A-DA-11 - Create interactive data visualizations using software tools to help others better understand real-world phenomena.
3B-AP-08 - Describe how artificial intelligence drives many software and physical systems.
3B-DA-05 - Use data analysis tools and techniques to identify patterns in data representing complex systems.
3B-DA-06 - Select data collection tools and techniques to generate data sets that support a claim or communicate information.

**Vision of A Learner Attributes:**
TCC1(9-12): I can ask purposeful, insightful questions to find a variety of innovative solutions.
TCC2(9-12): I can evaluate evidence from multiple perspectives, and recognize their limitations and implications, in order to justify new conclusions.

**Understandings:** Students will understand that...
- Data provide opportunities for identifying trends, making connections, and addressing problems.
- While computing innovations are typically designed to achieve a specific purpose, they may have unintended consequences.
- Responsible programmers try to consider the unintended ways their computing innovations can be used and the potential beneficial and harmful effects of these new uses.

**Essential Questions:**
- How can computation be employed to help people process data and information to gain insight and knowledge?
- When is it more appropriate to use a computer to analyze data than to complete the analysis by hand?
- Where and how does human bias affect the collection, processing and interpretation of data?

**Students will know…**
- Information is the collection of facts and patterns extracted from data.
- Digitally processed data may show correlation between variables. A correlation found in data does not necessarily indicate that a causal relationship exists. Additional research is needed to understand the exact nature of the relationship.
- Often, a single source does not contain the data needed to draw a conclusion. It may be necessary to combine data from a variety of sources to formulate a conclusion.
- Metadata are data about data. For example, the piece of data may be an image, while the metadata may include the date of creation or the file size of the image.

**Students will be able to…**
- 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.
- Describe what information can be extracted from data and metadata.
- Extract information from data using a program.
- Explain how programs can be used to gain insight and knowledge from data.
- Identify the challenges associated with processing data.
- Explain how people participate in problem-solving processes at scale.
- Explain how bias exists in computing innovations.

| | |
|---|---|
| • Changes and deletions made to metadata do not change the primary data.<br>• Search tools are useful for efficiently finding information.<br>• Tables, diagrams, text, and other visual tools can be used to communicate insight and knowledge gained from data.<br>• Programs such as spreadsheets help efficiently organize and find trends in information.<br>• Cleaning data is a process that makes the data uniform without changing its meaning (e.g., replacing all equivalent abbreviations, spellings, and capitalizations with the same word).<br>• Programmers can use programs to filter and clean digital data, thereby gaining insight and knowledge.<br>• Combining data sources, clustering data, and classifying data are parts of the process of using programs to gain insight and knowledge from data.<br>• Scalability of systems is an important consideration when working with data sets, as the computational capacity of a system affects how data sets can be processed and stored.<br>• Crowdsourcing is the practice of obtaining input or information from a large number of people via the Internet.<br>• Problems of bias are often created by the type or source of data being collected. Bias is not eliminated by simply collecting more data. | |
| **Key Vocabulary:** Citizen Science, Cleaning Data, Correlation, Crowdsourcing, Information, Data Bias, Data Filtering | |
| **Assessment Evidence** | |
| **Performance Tasks:**<br>Project Tell a Data Story:<br>The goal of this lesson is for students to put into use all of the data analysis skills they have practiced throughout this unit. | **Other Evidence:**<br>At the conclusion of this unit, students will complete a multiple choice assessment of AP-like problems to both assess their understanding of core concepts and to prepare them for the AP Exam they will take in May. |
| **Learning Plan** | |

| |
|---|
| In this unit, students learn how data analysis helps turn raw data into useful information about the world. They further learn how to use data visualization to find patterns inside of data sets and how this data analysis process is being used in contexts like open data or machine learning to help make decisions or learn more about our world. In the unit project, they analyze a dataset of their choosing and present their findings. |
| **Teacher Resources:**<br>Code.org AP CSP Online Materials - https://studio.code.org/courses/csp-2022?section_id=3618538<br>Supplemental Resources:<br>http://chortle.ccsu.edu/Java5/index.html#07<br>Java Runtime Environment<br>Java Development Kit |

# Unit 9 - Cybersecurity and Global Impacts

## Desired Results - Goals, Transfer, Meaning, Acquisition

**Established Goals:**
2-IC-23 - Describe tradeoffs between allowing information to be public and keeping information private and secure.
3A-IC-24 - Evaluate the ways computing impacts personal, ethical, social, economic, and cultural practices.
3A-IC-27 - Use tools and methods for collaboration on a project to increase connectivity of people in different cultures and career fields.
3A-IC-29 - Explain the privacy concerns related to the collection and generation of data through automated processes that may not be evident to users.
3A-IC-30 - Evaluate the social and economic implications of privacy in the context of safety, law, or ethics.
3A-NI-05 - Give examples to illustrate how sensitive data can be affected by malware and other attacks.
3A-NI-06 - Recommend security measures to address various scenarios based on factors such as efficiency, feasibility, and ethical impacts.
3A-NI-07 - Compare various security measures, considering tradeoffs between the usability and security of a computer system.
3B-IC-25 - Evaluate computational artifacts to maximize their beneficial effects and minimize harmful effects on society.
3B-IC-28 - Debate laws and regulations that impact the development and use of software.
3B-NI-04 - Compare ways software developers protect devices and information from unauthorized access.

**Vision of A Learner Attributes:**
TCC1(9-12): I can ask purposeful, insightful questions to find a variety of innovative solutions.
TCC2(9-12): I can evaluate evidence from multiple perspectives, and recognize their limitations and implications, in order to justify new conclusions.
TCC4(9-12): I can integrate my learning to adapt to experiences in the classroom, career and life.
AA4(9-12): I can create opportunities to extend my learning by remaining open-minded in any situation.

**Understandings:** Students will understand that...
- The use of computing innovations may involve risks to your personal safety and identity.
- Responsible programmers try to consider the unintended ways their computing innovations can be used and the potential beneficial and harmful effects of these new uses.

**Essential Questions:**
- What app or computer software do you use most often and would have a hard time going without? How does this software solve a problem for you or benefit you?
- Are innovators responsible for the harmful effects of their computing innovations, even if those effects were unintentional? Why or why not?
- How is cyber security impacting the ever-increasing number of internet users?

**Students will know…**
- Once information is placed online, it is difficult to delete.
- Devices, websites, and networks can collect information about a

**Students will be able to…**
- Research and debate current events at the intersection of data, public policy, law, ethics, and societal impact.

| | |
|---|---|
| user's location.<br>• Not every effect of a computing innovation is anticipated in advance.<br>• As with any technology or medium, using computing to harm individuals or groups of people raises legal and ethical concerns.<br>• Phishing is a technique that attempts to trick a user into providing personal information. That personal information can then be used to access sensitive online resources, such as bank accounts and emails.<br>• Keylogging is the use of a program to record every keystroke made by a computer user in order to gain fraudulent access to passwords and other confidential information.<br>• A rogue access point is a wireless access point that gives unauthorized access to secure networks.<br>• Computer virus and malware scanning software can help protect a computing system against infection.<br>• A computer virus is a malicious program that can copy itself and gain access to a computer in an unauthorized way. Computer viruses often attach themselves to legitimate programs and start running independently on a computer.<br>• Authentication measures protect devices and information from unauthorized access. Examples of authentication measures include strong passwords and multifactor authentication. | • Describe the risks to privacy from collecting and storing personal data on a computer system.<br>• Describe tradeoffs between allowing information to be public and keeping information private and secure.<br>• Evaluate the social and economic implications of privacy in the context of safety, law, or ethics.<br>• Explain how an effect of a computing innovation can be both beneficial and harmful.<br>• Explain how the use of computing could raise legal and ethical concerns.<br>• Explain how unauthorized access to computing resources is gained. |
| **Key Vocabulary:** Computing Innovation, Personally Identifiable Information (PIN) Phishing, Keylogging, Malware, Rogue Access Point, Encryption, Decryption, Symmetric Key Encryption, Public Key Encryption, Multi-factor Authentication | |
| **Assessment Evidence** | |
| **Performance Tasks:**<br>Project Innovation Simulation:<br>In this unit students will cover information related to computing innovations needed for the Curricular Requirement for the AP Computer Science Principles exam. This lesson sets the stage for the simulation. | **Other Evidence:**<br>At the conclusion of this unit, students will complete a multiple choice assessment of AP-like problems to both assess their understanding of core concepts and to prepare them for the AP Exam they will take in May. |

| Learning Plan |
| --- |
| In this unit, students learn how computing innovations have impacted our world in beneficial and harmful ways. They learn how data can pose a threat to our privacy and security and the ways that encryption and other techniques are used to protect it. Throughout the unit, students participate in a "school of the future" conference in which they and a team make a proposal for how best to improve school life with computing innovations. |
| **Teacher Resources:**<br>Code.org AP CSP Online Materials - https://studio.code.org/courses/csp-2022?section_id=3618538<br>Supplemental Resources:<br>http://chortle.ccsu.edu/Java5/index.html#07<br>Java Runtime Environment<br>Java Development Kit |