Wall Township Public Schools	Board Approval Date:
Course Title: AP Computer Science Principles	
Course Description: AP Computer Science Principles is an introductory college-level computing course. Students cultivate their understanding of computer science through working with data, collaborating to solve problems, and developing computer programs as they explore concepts like creativity, abstraction, data and information, algorithms, programming, the internet, and the global impact of computing. All students enrolled in Advanced Placement courses are required to take the Advanced Placement Examinations in May (see Policy 2429 & Program of Studies page 5).	

New Jersey Student Learning Standards
• 2020 New Jersey Student Learning Standards – Computer Science and Design Thinking Introduction
• 2020 New Jersey Student Learning Standards – Career Readiness, Life Literacies, and Key Skills

Course Sequence & Pacing

Unit Title	Estimated

	Number of Instructional Periods
Unit 1: Introduction to Programming and Computational Thinking (includes summer assignment)	6
Unit 2: JavaScript Programming Fundamentals	22
Unit 3: Data Structures and Algorithms	16
Unit 4: Digital Information and Data Representation	18
Unit 5: The Internet, Networks, and Cybersecurity	25
Unit 6: Data Analysis and Visualization	10
Unit 7: Create Performance Task and Exam Review	18
Unit 8: Creative Development Final Projects	20

Unit 1

Desired Results

Unit 1 Title: Introduction to Programming and Computational Thinking

Unit Summary:

This course begins with a strong focus on programming in order to allow students to create computational artifacts early on in the course. Students will be able to use their knowledge of programming to explore future topics in the course. We use Karel, a dog that only knows how to move, turn left, and place tennis balls in his world, to show students what it means to program, and allow students to focus on computational problem-solving. Students will learn about the

need for programming languages, the uses of programs, how to write programs to solve computational problems, how to design algorithms, how to analyze and compare potential solutions to programming problems, and learn the value and challenges involved in collaborating with others to solve programming problems.

College Board AP CSP Big Ideas covered in Unit 1: AAP, CRD

Learning Standards

NJSLS:

- 8.1.12.CS.4: Develop guidelines that convey systematic troubleshooting strategies that others can use to identify and fix errors.
- 8.1.12.AP.1: Design algorithms to solve computational problems using a combination of original and existing algorithms.
- 8.1.12.AP.3: Select and combine control structures for a specific application based upon performance and readability, and identify trade-offs to justify the choice.
- 8.1.12.AP.4: Design and iteratively develop computational artifacts for practical intent, personal expression, or to address a societal issue.
- 8.1.12.AP.5: Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects.
- 8.1.12.AP.6: Create artifacts by using procedures within a program, combinations of data and procedures, or independent but interrelated programs.
- 8.1.12.AP.7: Collaboratively design and develop programs and artifacts for broad audiences by incorporating feedback from users.
- 8.1.12.AP.9: Collaboratively document and present design decisions in the development of complex programs.
- 8.2.12.ED.1: Use research to design and create a product or system that addresses a problem and make modifications based on input from potential consumers.
- 8.2.12.NT.1: Explain how different groups can contribute to the overall design of a product.
- 9.4.12.CI.1: Demonstrate the ability to reflect, analyze, and use creative skills and ideas.
- 9.4.12.CT.1: Identify problem-solving strategies used in the development of an innovative product or practice.

- 9.4.12.CT.2: Explain the potential benefits of collaborating to enhance critical thinking and problem solving.
- 9.4.12.TL.1: Assess digital tools based on features such as accessibility options, capacities, and utility for accomplishing a specified task.
- 9.4.12.TL.3: Analyze the effectiveness of the process and quality of collaborative environments.

Interdisciplinary Connections

Career Readiness, Life Literacies, & Key Skills *(CLKS)*:

- 9.4.12.CI.1: Demonstrate the ability to reflect, analyze, and use creative skills and ideas.
- 9.4.12.CT.1: Identify problem-solving strategies used in the development of an innovative product or practice.
- 9.4.12.CT.2: Explain the potential benefits of collaborating to enhance critical thinking and problem solving.
- 9.4.12.TL.1: Assess digital tools based on features such as accessibility options, capacities, and utility for accomplishing a specified task.
- 9.4.12.TL.3: Analyze the effectiveness of the process and quality of collaborative environments.

Interdisciplinary Connections:

Science and Engineering Practices:

- Develop and use a model based on evidence to illustrate the relationships between systems or between components of a system.
- Create a computational model or simulation of a phenomenon, designed device, process, or system.
- Apply scientific principles and evidence to provide an explanation of phenomena and solve design problems, taking into account possible unanticipated effects.
- Analyze data using tools, technologies, and/or models (e.g., computational, mathematical) in order to make valid and reliable scientific claims or determine an optimal design solution.
- Apply scientific ideas to solve a design problem, taking into account possible unanticipated effects.
- Design, evaluate, and/or refine a solution to a complex real-world problem, based on scientific knowledge, student-generated sources of evidence, prioritized criteria, and tradeoff considerations.

- Use mathematical and/or computational representations of phenomena or design solutions to support explanations.
- Ask questions that arise from examining models or a theory to clarify relationships.
- Apply scientific reasoning to link evidence to the claims to assess the extent to which the reasoning and data support the explanation or conclusion.
- Construct an oral and written argument or counter-arguments based on data and evidence.

Engineering Design Standards:

- HS-ETS1-1 Analyze a major global challenge to specify qualitative and quantitative criteria and constraints for solutions that account for societal needs and wants.
- HS-ETS1-2 Design a solution to a complex real-world problem by breaking it down into smaller, more manageable problems that can be solved through engineering.
- HS-ETS1-3 Evaluate a solution to a complex real-world problem based on prioritized criteria and trade-offs that account for a range of constraints, including cost, safety, reliability, and aesthetics, as well as possible social, cultural, and environmental impacts.
- HS-ETS1-4 Use a computer simulation to model the impact of proposed solutions to a complex real-world problem with numerous criteria and constraints on interactions within and between systems relevant to the problem.

Learning Targets

Unit Essential Questions:

- How can you break big problems down into smaller problems?
- Are there subtasks that occur more than once in larger problems?
- What makes code readable and maintainable for yourself and others?

Unit Enduring Understandings:

- Decomposition is a fundamental problem-solving strategy.
- Functions and abstraction help manage complexity.
- Control structures enable flexible and dynamic solutions.

• How do control structures help create flexible solutions that work in different situations? • When should you use functions versus writing code directly in the main program?	• Readable code with proper structure is essential for collaboration and maintenance. • Algorithmic thinking skills transfer beyond programming.
Vocabulary: define a function, call a function, function, function body, lowerCamelCase, start function, top down design, decomposition, comments, precondition, postcondition, abstraction, API, documentation, for loop, if statement, conditional, if/else statement, while loop, control structure, fencepost problem	
Students will be able to: • Identify and brainstorm ways computers have impacted their own lives, as well as society, economy, and culture • Develop their own learning objectives for the course by reflecting on why they are taking this course • Recognize, explain, and use the commands that Karel can be given. These commands are: move(), putBall(), takeBall() and turnLeft() • Recognize, explain, and use the commands that Karel can be given • Debug and rework their code as well as someone else's code • Define and call functions • Understand what functions are, how they are used and how using them improves programs • Design and implement their own functions to solve problems • Explain the importance of writing readable code • Analyze and compare the readability of different programs • Use the start function to make their programs more readable • Break a large problem down into smaller, simpler problems • Write functions that solve the simpler problems, and use them as building blocks to solve the larger problem • Compare programs and identify good vs. poor decomposition • Explain the preconditions and postconditions of a function • Create clear and readable comments in their code that help the reader understand the code • Explain the purpose of comments	

- Understand abstraction as the different levels of detail and complexity
- Understand the importance of abstracting away complexity to solve problems more efficiently
- Write programs that use SuperKarel instead of Karel
- Utilize the new toolbox of commands that SuperKarel provides over Karel
- Read the documentation to understand how to use an API (SuperKarel is an example of this)
- Create for loops to repeat code a fixed number of times
- Explain when a for loop would be a useful tool
- Utilize for loops to write programs that would be difficult/impossible without loops
- Use conditions to gather information about Karel's world (is the front clear, is Karel facing north, etc)
- Create if statements to only execute code if a certain condition is true
- Explain the purpose of an If/Else statement
- Create If/Else statements to solve new types of problems
- Identify when an If/Else statement is appropriate to be used
- Explain the purpose of a while loop
- Create while loops to repeat code while a condition is true
- Utilize while loops to solve new types of problems
- Test their solutions on different Karel worlds
- Identify the different control structures that can be used to modify the flow of control through a program
- Combine control structures to solve complicated problems
- Choose the proper control structure for a given problem
- Students will be able to use debugging strategies to find and fix errors in their code.
- Analyze an algorithm and explain why it works
- Use control structures to create general algorithms that work on all Karel worlds
- Explain why it is important to indent code
- Identify proper indentation
- Modify a program to have proper indentation
- Write programs with proper indentation

- Use Ultra Karel commands to paint Karel's world
- Call functions that accept parameters as inputs
- Explain the relationship between a function and a parameter
- Create generalized Ultra Karel algorithms that correctly solve multiple Karel worlds
- Identify differences between the Super Karel API and the Ultra Karel API
- Define a problem in their own words and plan out a solution to the problem
- Break a large problem down into smaller pieces and solve each of the pieces, then use these solutions as building blocks to solve the larger problem
- Utilize the proper control structures to create general solutions that solve multiple Karel worlds
- Write clear and readable code using control structures, functions, decomposition, and comments
- Prove their knowledge of control structures, functions, decomposition, and comments through a multiple choice quiz
- Develop a program for creative expression, or to satisfy personal curiosity
- Develop an algorithm and implement it using Karel's JavaScript library of commands and conditions
- Collaborate in the development of programs
- Make a plan for their solution by breaking their problem down into solvable subproblems using Top Down Design
- Develop abstractions to manage the complexity of their program (write several reusable functions that solve small subproblems)

Learning Activities & Differentiation: *(Amistad & Holocaust, Asian American & Pacific Islanders, Climate Change, LGBTQ and People with Disabilities, Diversity, Equity & Inclusion [DEI])*

Learning Activities:

- Interactive Karel programming exercises progressing from basic commands to complex problem-solving
- Pair programming sessions where students collaborate on Karel challenges
- Code debugging workshops using peer review and systematic debugging strategies
- Function design workshops with real-world problem decomposition exercises
- Algorithm analysis activities comparing different solutions to the same Karel problem

- Creative expression projects where students design their own Karel worlds and solutions

Differentiation Strategies:

- For Advanced Learners: Provide extension challenges with multiple solution paths, mentor struggling peers, create original Karel worlds with custom challenges
- For Struggling Learners: Offer step-by-step guided practice, visual flowcharts for algorithm planning, one-on-one coding support sessions
- For English Language Learners: Use visual programming aids, provide glossary of programming terms, allow code comments in native language initially
- Universal Design for Learning: Multiple representation formats (visual, kinesthetic, auditory), choice in final project topics, flexible pacing for concept mastery

Education Connections:

- Highlight diverse programmers and computer scientists in historical contexts
- Discuss how programming accessibility features help people with disabilities
- Explore how computational thinking applies across cultures and communities
- Address equity in technology access and opportunities

Assessment Evidence

Formative Assessments:

- CodeHS coding exercises
    - Example exercise: The Two Towers

      In this program, students have Karel build two towers of tennis balls. Each tower should be 3 tennis balls high. In the end, Karel should end up on top of the second tower, facing East. Students need to write at least 3 functions in order to solve this problem. This activity requires students to design and create functions for repeated processes within their program.
- CodeHS debugging exercises

- CodeHS Checks for Understanding

Summative Assessments:

- Karel Challenges
- Practice PT: Pair Programming Paint!

Common Benchmark Assessments:

- Karel Challenges
- Practice PT: Pair Programming Paint!

Alternative Assessments:

- CodeHS Unit Quiz

Core Instructional & Supplemental Materials

Core Materials:

- CodeHS Karel programming platform and curriculum
- Karel JavaScript library and development environment
- Interactive debugging tools and error analysis resources
- Function documentation and API reference guides

Supplemental Materials:

- Visual flowchart templates for algorithm design
- Peer collaboration rubrics for pair programming
- Real-world problem-solving case studies
- Assessment rubrics for creative projects

Accommodations and Modifications:
Students with Disabilities, English Language Learners, Students at Risk of Failure, Students with 504s, Gifted & Talented Students

Specific Strategies and Practices that Support Students with Disabilities:
Students receiving Special Education programming have specific goals and objectives, as well as accommodations and modifications outlined within their Individualized Education Plans (IEP) due to an identified disability and/or diagnosis. In addition to exposure to the general education curriculum, the instruction is differentiated based upon the student's needs. The IEP acts as a supplemental curriculum guide inclusive of instructional strategies that support each learner.

- Additional time - Review of directions - Space for movement or breaks - Extra visual and verbal cues and prompts - Follow a routine/schedule - Verbal and visual cues regarding directions and staying on task - Checklists - Immediate feedback

Specific Strategies and Practices that Support English Language Learners
English language learner (ELL) is a national-origin-minority student who is limited-English-proficient. This term is often preferred over limited-English-proficient (LEP) as it highlights accomplishments rather than deficits.

- Pre-teaching of vocabulary and concepts - Personal Glossary - Extended Time - Visual learning, including graphic organizers - Teacher modeling - Pairing students with beginning English language skills with students who have more advanced English language skills - Think-pair-share - Cooperative learning groups

Students at Risk of Failure

- Alternative Assessments - Learning Contracts - Supplemental Instruction - Think-Pair-Share by Readiness - Differentiated Online Practice - Tiered Level of Supports - Tiered Activities - Small Group Instruction - Work Alone Choice - Group Work Choice

Students with 504 Plans
Teachers are responsible for implementing designated services and strategies identified on a student’s 504 Plan.
504 plans are formal plans that schools develop to give kids the support they need and covers any condition that limits daily activities in a major way. Students' accommodations are outlined in their individual 504 plan.

Specific Strategies and Practices that Support Gifted & Talented Students
It is critical that all teachers are able to recognize a high-ability student who may need more depth and complexity in instruction.

- Small Group In-Depth Instruction - Differentiated Online Practice - Alternative Assessments - Pre-Assess to provide Differentiated Instruction to Accelerate Learning

Unit 2

Desired Results

Unit 2 Title: JavaScript Programming Fundamentals

Unit Summary:

This unit introduces students to the basics of JavaScript, including variables, user input, control structures, functions with parameters and return values, and basic graphics, how to send messages to objects. Students also learn how to use booleans and logical operators with control structures to make more advanced programs in JavaScript.

College Board AP CSP Big Ideas covered in Unit 1: AAP, CRD

Learning Standards

NJSLS:

- 8.1.12.CS.1 - Describe ways in which integrated systems hide underlying implementation details to simplify user experiences
- 8.1.12.CS.4 - Develop guidelines that convey systematic troubleshooting strategies that others can use to identify and fix errors
- 8.1.12.DA.3 - Translate between decimal numbers and binary numbers
- 8.1.12.DA.4 - Explain the relationship between binary numbers and the storage and use of data in a computing device
- 8.1.12.AP.1 - Design algorithms to solve computational problems using a combination of original and existing algorithms
- 8.1.12.AP.2 - Create generalized computational solutions using collections instead of repeatedly using simple variables
- 8.1.12.AP.3 - Select and combine control structures for a specific application based upon performance and readability

- 8.1.12.AP.4 - Design and iteratively develop computational artifacts for practical intent, personal expression, or to address a societal issue
- 8.1.12.AP.5 - Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects
- 8.1.12.AP.6 - Create artifacts by using procedures within a program, combinations of data and procedures, or independent but interrelated programs
- 8.1.12.AP.7 - Collaboratively design and develop programs and artifacts for broad audiences by incorporating feedback from users
- 8.1.12.IC.1 - Evaluate the ways computing impacts personal, ethical, social, economic, and cultural practices
- 8.2.12.ED.1 - Use research to design and create a product or system that addresses a problem and make modifications based on input from potential consumers
- 8.2.12.ED.4 - Design a product or system that addresses a global problem and document decisions made based on research, constraints, trade-offs, and aesthetic and ethical considerations

Interdisciplinary Connections

Career Readiness, Life Literacies, & Key Skills *(CLKS)*:

- 9.4.12.CT.1 - Identify problem-solving strategies used in the development of an innovative product or practice
- 9.4.12.CT.2 - Explain the potential benefits of collaborating to enhance critical thinking and problem solving
- 9.4.12.DC.6 - Select information to post online that positively impacts personal image and future college and career opportunities
- 9.4.12.DC.7 - Evaluate the influence of digital communities on the nature, content and responsibilities of careers
- 9.4.12.IML.1 - Compare search browsers and recognize features that allow for filtering of information
- 9.4.12.IML.2 - Evaluate digital sources for timeliness, accuracy, perspective, credibility of the source, and relevance of information
- 9.4.12.IML.3 - Analyze data using tools and models to make valid and reliable claims
- 9.4.12.TL.1 - Assess digital tools based on features such as accessibility options, capacities, and utility for accomplishing a specified task
- 9.4.12.TL.2 - Generate data using formula-based calculations in a spreadsheet and draw conclusions about the data
- 9.4.12.TL.3 - Analyze the effectiveness of the process and quality of collaborative environments

- 9.4.12.CI.1 - Demonstrate the ability to reflect, analyze, and use creative skills and ideas
- 9.4.12.CI.2 - Identify career pathways that highlight personal talents, skills, and abilities

Interdisciplinary Connections:

Science and Engineering Practices:

- Develop a model based on evidence to illustrate the relationships between systems or between components of a system.
- Use a model to predict the relationships between systems or between components of a system.
- Analyze data using tools, technologies, and/or models (e.g., computational, mathematical) in order to make valid and reliable scientific claims or determine an optimal design solution.
- Use mathematical and/or computational representations of phenomena or design solutions to support explanations.
- Create a computational model or simulation of a phenomenon, designed device, process, or system.
- Apply scientific ideas to solve a design problem, taking into account possible unanticipated effects.
- Design, evaluate, and/or refine a solution to a complex real-world problem, based on scientific knowledge, student-generated sources of evidence, prioritized criteria, and tradeoff considerations.
- Evaluate the claims, evidence, and reasoning behind currently accepted explanations or solutions to determine the merits of arguments.
- Apply scientific reasoning to link evidence to the claims to assess the extent to which the reasoning and data support the explanation or conclusion.
- Construct an explanation based on valid and reliable evidence obtained from a variety of sources (including students' own investigations, models, theories, simulations, peer review) and the assumption that theories and laws that describe the natural world operate today as they did in the past and will continue to do so in the future.

Engineering Design Standards:

- HS-ETS1-2 Design a solution to a complex real-world problem by breaking it down into smaller, more manageable problems that can be solved through engineering.

- HS-ETS1-3 Evaluate a solution to a complex real-world problem based on prioritized criteria and trade-offs that account for a range of constraints, including cost, safety, reliability, and aesthetics, as well as possible social, cultural, and environmental impacts.
- HS-ETS1-4 Use a computer simulation to model the impact of proposed solutions to a complex real-world problem with numerous criteria and constraints on interactions within and between systems relevant to the problem.

Learning Targets

Unit Essential Questions:

- How do programming languages enable humans to communicate instructions to computers?
- What makes a program effective at solving real-world problems?
- How do variables serve as abstractions that simplify complex data management?
- How do programs make decisions and adapt their behavior based on different conditions?
- How can iteration structures efficiently solve problems that involve repetitive tasks?

Unit Enduring Understandings:

- Programming languages provide a structured way to express algorithms and solve computational problems.
- Variables act as named storage locations that abstract away the complexity of memory management.
- Selection structures (if/else statements) allow programs to execute different code paths based on conditions
- Iteration structures (loops) enable programs to repeat actions efficiently without code duplication.
- Interactive programs use events and user input to create dynamic, responsive experiences that can adapt to different situations and handle uncertainty.

Vocabulary: code, programming language, algorithm, Hello World, println, boolean, variable, declare, initialize, readLine, readInt, readFloat, increment, decrement, arithmetic operators, canvas, event, callback function, lowerCamelCase, input,

output, getWidth(), getHeight(), logical operator, comparison operator, if statement, if/else statement, for loop, iteration, randomize, pseudorandom, while loop, infinite loop, sentinel, break, loop-and-a-half
Students will be able to: • Students will be able to explain what code is in their own words, and provide examples of code in their lives. • Students understand why programming is a useful skill, and can explain ways in which programs are being used today. Students will be able to analyze the positive and negative effects of programs and communicate their findings to their classmates. • Write a JavaScript program by typing commands with proper syntax in the start function • Write a program that prints out a message to the user • Explain what variables are and what they are used for • Use the different types of variables in JavaScript • Distinguish between declaring, initializing and assigning variables • Create their own variables with proper naming conventions • Print out the values stored in variables • Create programs that ask the user for input • Store user input in variables and print it back to the user • Choose the proper input function to use depending on the type of information needed • Describe the different mathematical operators we can use in programs • Create programs that use basic math to compute useful things • Create programs that take in user input, do simple computations with the input, and produce useful output • Create graphical JavaScript programs that draw shapes on the canvas • Locate points on the graphics canvas using (x, y) coordinates • Describe how events are different than timers • Use mouse click events to create programs that respond to user clicks • Create boolean variables to represent meaningful yes/no values • Print out the value of a boolean variable • Describe the meaning and usage of each logical operator: OR (||), AND (&&), and NOT (!)

- Construct logical statements using boolean variables and logical operators
- Explain the meaning of each of the comparison operators (<, <=, >, >=, ==, !=)
- Create programs using the comparison operators to compare values
- Predict the boolean result of comparing two values
- Print out the boolean result of comparing values
- Explain the purpose of if statements
- Create their own if statements to selective choose which code is executed in their programs
- Create interactive programs that use events to respond to the keyboard input.
- Create for loops in JavaScript
- Explain the purpose of for loops
- Utilize for loops to avoid typing out repeated code
- Use the loop counter i inside the for loop code to do something different on each iteration
- Explain the three parts of the for loop (initialization statement, test statement, increment statement)
- Create for loops that iterate differently than the basic for loop structure (ie count by twos or count backwards)
- Explain the purpose of for loops
- Create for loops to solve increasingly challenging problems
- Create nested for loops
- Explain why random numbers are a useful part of computer programs
- Create random values in a program
- Utilize the DOCS for the Randomizer class in order to learn how to generate random values
- Explain the purpose of a while loop
- Create while loops to repeat code while a condition is true
- Utilize while loops to solve new types of problems
- Explain how the loop-and-a-half structure is different from a traditional while loop
- Explain what an infinite loop is
- Explain what the break statement does

- Create programs that use the loop-and-a-half structure to repeat code until a SENTINEL is met, causing the program to break out of the loop

Learning Activities & Differentiation: *(Amistad & Holocaust, Asian American & Pacific Islanders, Climate Change, LGBTQ and People with Disabilities, Diversity, Equity & Inclusion [DEI])*

Learning Activities:

- Interactive JavaScript variable and input exercises progressing from basic declarations to complex user interaction systems
- Collaborative graphics programming projects where students create canvas-based narratives using coordinate systems and events
- Code review sessions focusing on control structure design and boolean logic implementation
- Loop construction workshops with real-world problem-solving scenarios involving iteration and repetition
- Conditional logic comparison activities analyzing different approaches to decision-making structures
- Creative programming projects using randomization and interactive simulations for data modeling

Differentiation Strategies:

- For Advanced Learners: Provide complex nested loop challenges, mentor peers in debugging, create original algorithms for graphics programming
- For Struggling Learners: Offer scaffolded code templates, visual flowcharts for control structures, guided practice with variable usage
- For English Language Learners: Use visual programming concepts, provide multilingual programming glossaries, allow gradual transition to English comments
- Universal Design for Learning: Multiple coding environments, choice in project themes, flexible assessment formats, varied presentation modes

Education Connections:

- Highlight diverse contributions to programming languages and computational thinking across cultures

- Discuss how interactive programs support accessibility technologies and assistive devices
- Explore how loops and variables model climate change and environmental data from global perspectives
- Address equity in user input design and algorithmic decision-making in real-world applications

Assessment Evidence

Formative Assessments:

- CodeHS coding exercises
- CodeHS debugging exercises
- CodeHS Checks for Understanding

Summative Assessments:

- Click for Rectangles (CodeHS)

Common Benchmark Assessments:

- Click for Rectangles (CodeHS)

Alternative Assessments:

- CodeHS Unit Quiz

Core Instructional & Supplemental Materials

Core Materials:

- CodeHS JavaScript development environment and interactive coding platform
- Built-in JavaScript functions and method documentation (DOCS tab)
- Interactive debugging tools and error analysis resources
- Control structure templates and syntax reference guides

Supplemental Materials:

- Visual flowchart templates for algorithm and conditional logic design

- Peer collaboration rubrics for pair programming and code review
- Real-world problem-solving case studies featuring loops and decision-making
- Assessment rubrics for interactive programming projects

Accommodations and Modifications: Students with Disabilities, English Language Learners, Students at Risk of Failure, Students with 504s, Gifted & Talented Students

Specific Strategies and Practices that Support Students with Disabilities:

Students receiving Special Education programming have specific goals and objectives, as well as accommodations and modifications outlined within their Individualized Education Plans (IEP) due to an identified disability and/or diagnosis. In addition to exposure to the general education curriculum, the instruction is differentiated based upon the student's needs. The IEP acts as a supplemental curriculum guide inclusive of instructional strategies that support each learner.

- Student restates information - Extra visual and verbal cues and prompts - Follow a routine/schedule - Checklists - Immediate feedback

Specific Strategies and Practices that Support English Language Learners

English language learner (ELL) is a national-origin-minority student who is limited-English-proficient. This term is often preferred over limited-English-proficient (LEP) as it highlights accomplishments rather than deficits.

- Pre-teaching of vocabulary and concepts - Personal Glossary - Visual learning, including graphic organizers - Teacher modeling - Pairing students with beginning English language skills with students who have more advanced English language skills

Students at Risk of Failure

- Alternative Assessments - Think-Pair-Share by Readiness - Homework Options - Differentiated Online Practice - Small Group Instruction - Group Work Choice

Students with 504 Plans

Teachers are responsible for implementing designated services and strategies identified on a student’s 504 Plan.

504 plans are formal plans that schools develop to give kids the support they need and covers any condition that limits daily activities in a major way. Students' accommodations are outlined in their individual 504 plan.

Specific Strategies and Practices that Support Gifted & Talented Students
It is critical that all teachers are able to recognize a high-ability student who may need more depth and complexity in instruction.
-Homogenous Grouping - Alternative Assessments

Unit 3

Desired Results

Unit 3 Title: Data Structures and Algorithms

Unit Summary:

In this unit, students learn to write reusable code with functions and parameters. Students will write a JavaScript program that tells a graphical story. Students will also learn about lists and arrays which are essential basic data structures that any program will use.

College Board AP CSP Big Ideas covered in Unit 1: AAP, DAT

Learning Standards

NJSLS:

- 8.1.12.DA.1 - Create interactive data visualizations using software tools to help others better understand real world phenomena, including climate change
- 8.1.12.DA.5 - Create data visualizations from large data sets to summarize, communicate, and support different interpretations of real-world phenomena
- 8.1.12.DA.6 - Create and refine computational models to better represent the relationships among different elements of data collected from a phenomenon or process

- 8.1.12.AP.1 - Design algorithms to solve computational problems using a combination of original and existing algorithms
- 8.1.12.AP.2 - Create generalized computational solutions using collections instead of repeatedly using simple variables
- 8.1.12.AP.3 - Select and combine control structures for a specific application based upon performance and readability, and identify trade-offs to justify the choice
- 8.1.12.AP.4 - Design and iteratively develop computational artifacts for practical intent, personal expression, or to address a societal issue
- 8.1.12.AP.5 - Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects
- 8.1.12.AP.6 - Create artifacts by using procedures within a program, combinations of data and procedures, or independent but interrelated programs
- 8.2.12.ED.1 - Use research to design and create a product or system that addresses a problem and make modifications based on input from potential consumers

Interdisciplinary Connections

Career Readiness, Life Literacies, & Key Skills *(CLKS)*:

- 9.4.12.CI.1 – Demonstrate the ability to reflect, analyze, and use creative skills and ideas
- 9.4.12.CI.2 – Identify career pathways that highlight personal talents, skills, and abilities
- 9.4.12.CT.1 – Identify problem-solving strategies used in the development of an innovative product or practice
- 9.4.12.CT.2 – Explain the potential benefits of collaborating to enhance critical thinking and problem solving
- 9.4.12.CT.4 – Participate in online strategy and planning sessions for course-based, school-based, or other project and determine the strategies that contribute to effective outcomes
- 9.4.12.DC.1 – Explain the beneficial and harmful effects that intellectual property laws can have on the creation and sharing of content
- 9.4.12.DC.3 – Evaluate the social and economic implications of privacy in the context of safety, law, or ethics
- 9.4.12.DC.4 – Explain the privacy concerns related to the collection of data (e.g., cookies) and generation of data through automated processes that may not be evident to users (e.g., 8.1.12.NI.3).

- 9.4.12.GCA.1 – Collaborate with individuals to analyze a variety of potential solutions to climate change effects and determine why some solutions may work better than others
- 9.4.12.IML.3 – Analyze data using tools and models to make valid and reliable claims, or to determine optimal design solutions
- 9.4.12.IML.4 – Assess and critique the appropriateness and impact of existing data visualizations for an intended audience
- 9.4.12.TL.1 – Assess digital tools based on features such as accessibility options, capacities, and utility for accomplishing a specified task
- 9.4.12.TL.3 – Analyze the effectiveness of the process and quality of collaborative environments
- 9.4.12.TL.4 – Collaborate in online learning communities or social networks or virtual worlds to analyze and propose a resolution to a real-world problem

Interdisciplinary Connections:

Science and Engineering Practices:

- Develop and use a model based on evidence to illustrate the relationships between systems or between components of a system.
- Create a computational model or simulation of a phenomenon, designed device, process, or system.
- Use mathematical and/or computational representations of phenomena or design solutions to support explanations.
- Design, evaluate, and/or refine a solution to a complex real-world problem, based on scientific knowledge, student-generated sources of evidence, prioritized criteria, and tradeoff considerations.
- Apply scientific reasoning to link evidence to the claims to assess the extent to which the reasoning and data support the explanation or conclusion.

Engineering Design Standards:

- HS-ETS1-2 Design a solution to a complex real-world problem by breaking it down into smaller, more manageable problems that can be solved through engineering.

- HS-ETS1-4 Use a computer simulation to model the impact of proposed solutions to a complex real-world problem with numerous criteria and constraints on interactions within and between systems relevant to the problem.

Learning Targets

Unit Essential Questions:

- How do functions and parameters enable code reusability and abstraction in programming?
- What makes data structures essential tools for organizing and managing information in programs?
- How do algorithms efficiently process collections of data through iteration and traversal?
- In what ways do simulations use data structures and algorithms to model real-world phenomena?
- How can programs be designed to search, filter, and transform data to solve complex problems?

Unit Enduring Understandings:

- Functions with parameters provide abstraction by encapsulating reusable code that can operate on different inputs.
- Data structures like arrays and lists organize collections of related data and enable efficient storage and retrieval.
- Iteration and traversal algorithms allow programs to systematically process each element in a data structure.
- Simulations use mathematical models, random number generation, and data structures to represent real-world systems.
- Search and filtering algorithms transform raw data into meaningful information through systematic processing techniques.

Vocabulary: function, parameter, return, scope, local variable, API, global variable, array, index, push, pop, traverse an array, iterate, spice, remove, simulation, model, data structure,

Students will be able to:

- Explain the purpose of functions
- Define their own JavaScript functions

- Utilize (call) their JavaScript functions to solve simple problems
- Define and call functions that take in parameters as input
- Identify repeated code that can be simplified with functions and parameters
- Define functions that take in multiple parameters as input, and create graphics as output
- Pass parameters of the correct number and type to their defined JavaScript functions
- Explain the purpose of returning a value from a function.
- Define functions that return values.
- Create programs that define and call functions with return values and store the result for later use.
- Explain the purpose of returning a value from a function.
- Create functions that return values.
- Create programs that call functions with return values and use the return values to solve a higher-order problem.
- Identify the scope of a variable
- Identify which variables are in scope at a given point in a program
- Understand how an API abstracts away detail for a program such as Karel
- Use functions and event-driven programming to create a simple API
- Synthesize the skills and concepts from the JavaScript and Graphics, JavaScript Control Structures, and the Functions and Parameters units to solve increasingly difficult programming challenges
- Break down a large problem into smaller parts using Top Down Design, and solve each of these smaller parts using functions
- Create helpful comments with preconditions and postconditions to help the reader understand the code
- Find and fix bugs in large programs
- Define an array
- Access certain elements of an array by using an index
- Use indexing to call and assign items in an array
- Add elements at the end of an array using the push method
- Remove elements from the end of an array using the pop method
- Determine the length of an array using the length property

- Use the length of an array and a for loop to loop through the elements in an array
- Loop over an array to filter or print certain elements based on tested criteria
- Use the length of an array and a for loop to loop through the elements in an array
- Loop over an array to filter or print certain elements based on tested criteria
- Using iteration on arrays for problems involving randomness and graphics
- Use the indexOf method to find the index of a particular element in an array.
- Use the splice and remove methods to remove an element from an array.

Learning Activities & Differentiation: *(Amistad & Holocaust, Asian American & Pacific Islanders, Climate Change, LGBTQ and People with Disabilities, Diversity, Equity & Inclusion [DEI])*

Learning Activities:

- Interactive JavaScript function creation exercises progressing from basic parameters to complex return values
- Collaborative storytelling projects where students use functions to create graphical narratives
- Code review sessions focusing on function design and data structure implementation
- Array manipulation workshops with real-world data processing scenarios
- Algorithm comparison activities analyzing different approaches to searching and sorting
- Creative programming projects using data visualization and interactive simulations

Differentiation Strategies:

- For Advanced Learners: Provide complex data structure challenges, mentor peers in debugging, create original algorithms for data processing
- For Struggling Learners: Offer scaffolded function templates, visual array representations, guided practice with parameter usage
- For English Language Learners: Use visual programming concepts, provide multilingual programming glossaries, allow gradual transition to English comments
- Universal Design for Learning: Multiple coding environments, choice in project themes, flexible assessment formats, varied presentation modes

Education Connections:

- Highlight diverse contributions to data science and algorithm development across cultures
- Discuss how data structures support accessibility technologies and assistive devices
- Explore how simulations model climate change and environmental data from global perspectives
- Address equity in data representation and algorithmic bias in real-world applications

Assessment Evidence

Formative Assessments:

- CodeHS coding exercises
    - Example Exercise: Better Password Prompt

      Students write a program that uses a while loop to prompt a user for a password. They keep prompting the user for the password, and if they get it correct, they then break out of the loop. If they don't get it correct, they should give the user an error message. This activity requires that students use multiple program statements in a specific order to solve a problem.
- CodeHS debugging exercises
- CodeHS Checks for Understanding

Summative Assessments:

- JavaScript and Graphics Challenges

Common Benchmark Assessments:

- JavaScript and Graphics Challenges

Alternative Assessments:

- CodeHS Unit Quiz

Core Instructional & Supplemental Materials

Core Materials:

- CodeHS JavaScript development environment and interactive coding platform
- Built-in JavaScript functions and method documentation (DOCS tab)
- Interactive debugging tools and error analysis resources
- Control structure templates and syntax reference guides

Supplemental Materials:

- Visual flowchart templates for algorithm and conditional logic design
- Peer collaboration rubrics for pair programming and code review
- Real-world problem-solving case studies featuring loops and decision-making
- Assessment rubrics for interactive programming projects

Accommodations and Modifications: Students with Disabilities, English Language Learners, Students at Risk of Failure, Students with 504s, Gifted & Talented Students

Specific Strategies and Practices that Support Students with Disabilities:

Students receiving Special Education programming have specific goals and objectives, as well as accommodations and modifications outlined within their Individualized Education Plans (IEP) due to an identified disability and/or diagnosis. In addition to exposure to the general education curriculum, the instruction is differentiated based upon the student's needs. The IEP acts as a supplemental curriculum guide inclusive of instructional strategies that support each learner.

- Small group/One to one - Additional time - Review of directions - Student restates information - Space for movement or breaks - Extra visual and verbal cues and prompts - Preferential seating - Follow a routine/schedule - Rest breaks - Verbal and visual cues regarding directions and staying on task - Checklists - Immediate feedback

Specific Strategies and Practices that Support English Language Learners

English language learner (ELL) is a national-origin-minority student who is limited-English-proficient. This term is often preferred over limited-English-proficient (LEP) as it highlights accomplishments rather than deficits.

- Pre-teaching of vocabulary and concepts - Personal Glossary - Extended Time - Simplified/Verbal Instructions - Frequent Breaks - Visual learning, including graphic organizers - Use of cognates to increase comprehension - Teacher modeling - Pairing students with beginning English language skills with students who have more advanced English language skills - Scaffolding - Word walls - Think-pair-share - Cooperative learning groups

Students at Risk of Failure

- Home/School Connection - Alternative Assessments - Learning Contracts - Homogenous Grouping - Supplemental Instruction - Think-Pair-Share by Readiness - Homework Options - Differentiated Online Practice - Leveled Rubrics - Tiered Level of Supports - Tiered Activities - Small Group Instruction - Work Alone Choice - Group Work Choice

Students with 504 Plans

Teachers are responsible for implementing designated services and strategies identified on a student's 504 Plan.
504 plans are formal plans that schools develop to give kids the support they need and covers any condition that limits daily activities in a major way. Students' accommodations are outlined in their individual 504 plan.

Specific Strategies and Practices that Support Gifted & Talented Students

It is critical that all teachers are able to recognize a high-ability student who may need more depth and complexity in instruction.

-Homogenous Grouping - Small Group In-Depth Instruction - Differentiated Online Practice - Alternative Assessments - Pre-Assess to provide Differentiated Instruction to Accelerate Learning

Unit 4

Desired Results

Unit 4 Title: Digital Information and Data Representation

Unit Summary:

In this unit, students will learn about the various ways we represent information digitally. Topics covered include number systems, encoding data, programmatically creating pixel images, comparing data encodings, compressing and encrypting data. Students will work in pairs to develop their own data encryption algorithms and attempt to crack the encryptions of their peers. Their text encryption tool will be embedded in their portfolio websites.

College Board AP CSP Big Ideas covered in Unit 1: DAT, IOC

Learning Standards

NJSLS:

- 8.1.12.CS.1: Describe ways in which integrated systems hide underlying implementation details to simplify user experiences.
- 8.1.12.CS.2: Model interactions between application software, system software, and hardware.
- 8.1.12.CS.3: Compare the functions of application software, system software, and hardware.
- 8.1.12.NI.2: Evaluate security measures to address various common security threats.
- 8.1.12.NI.3: Explain how the needs of users and the sensitivity of data determine the level of security implemented.
- 8.1.12.NI.4: Explain how decisions on methods to protect data are influenced by whether the data is at rest, in transit, or in use.
- 8.1.12.IC.1: Evaluate the ways computing impacts personal, ethical, social, economic, and cultural practices.
- 8.1.12.IC.2: Test and refine computational artifacts to reduce bias and equity deficits.
- 8.1.12.IC.3: Predict the potential impacts and implications of emerging technologies on larger social, economic, and political structures, using evidence from credible sources.
- 8.1.12.DA.1: Create interactive data visualizations using software tools to help others better understand real world phenomena, including climate change.
- 8.1.12.DA.2: Describe the trade-offs in how and where data is organized and stored.
- 8.1.12.DA.3: Translate between decimal numbers and binary numbers.
- 8.1.12.DA.4: Explain the relationship between binary numbers and the storage and use of data in a computing device.
- 8.1.12.DA.5: Create data visualizations from large data sets to summarize, communicate, and support different

interpretations of real-world phenomena.
- 8.1.12.DA.6: Create and refine computational models to better represent the relationships among different elements of data collected from a phenomenon or process.

Interdisciplinary Connections

Career Readiness, Life Literacies, & Key Skills *(CLKS)*:

- 9.4.12.CI.1: Demonstrate the ability to reflect, analyze, and use creative skills and ideas.
- 9.4.12.CI.2: Identify career pathways that highlight personal talents, skills, and abilities.
- 9.4.12.CT.1: Identify problem-solving strategies used in the development of an innovative product or practice.
- 9.4.12.CT.2: Explain the potential benefits of collaborating to enhance critical thinking and problem solving.
- 9.4.12.CT.4: Participate in online strategy and planning sessions for course-based, school-based, or other projects and determine the strategies that contribute to effective outcomes.
- 9.4.12.DC.1: Explain the beneficial and harmful effects that intellectual property laws can have on the creation and sharing of content.
- 9.4.12.DC.3: Evaluate the social and economic implications of privacy in the context of safety, law, or ethics.
- 9.4.12.DC.4: Explain the privacy concerns related to the collection of data and generation of data through automated processes that may not be evident to users.
- 9.4.12.IML.3: Analyze data using tools and models to make valid and reliable claims, or to determine optimal design solutions.
- 9.4.12.IML.4: Assess and critique the appropriateness and impact of existing data visualizations for an intended audience.
- 9.4.12.IML.7: Develop an argument to support a claim regarding a current workplace or societal/ethical issue such as climate change.
- 9.4.12.TL.1: Assess digital tools based on features such as accessibility options, capacities, and utility for accomplishing a specified task.
- 9.4.12.TL.3: Analyze the effectiveness of the process and quality of collaborative environments.

Interdisciplinary Connections:

Science and Engineering Practices:

- Develop and use a model based on evidence to illustrate the relationships between systems or between components of a system.
- Create a computational model or simulation of a phenomenon, designed device, process, or system.
- Design, evaluate, and/or refine a solution to a complex real-world problem, based on scientific knowledge, student-generated sources of evidence, prioritized criteria, and tradeoff considerations.
- Analyze data using tools, technologies, and/or models (e.g., computational, mathematical) in order to make valid and reliable scientific claims or determine an optimal design solution.
- Use mathematical and/or computational representations of phenomena or design solutions to support explanations.
- Apply scientific reasoning to link evidence to the claims to assess the extent to which the reasoning and data support the explanation or conclusion.

Engineering Design Standards:

- HS-ETS1-2 Design a solution to a complex real-world problem by breaking it down into smaller, more manageable problems that can be solved through engineering.
- HS-ETS1-3 Evaluate a solution to a complex real-world problem based on prioritized criteria and trade-offs that account for a range of constraints, including cost, safety, reliability, and aesthetics, as well as possible social, cultural, and environmental impacts.
- HS-ETS1-4 Use a computer simulation to model the impact of proposed solutions to a complex real-world problem with numerous criteria and constraints on interactions within and between systems relevant to the problem.

Learning Targets	
Unit Essential Questions:	**Unit Enduring Understandings:** • All digital data, including text, images, and sound, is ultimately represented using binary, and this

<table>
<tr>
<td>

- How is digital information represented and manipulated using number systems like binary and hexadecimal?
- What trade-offs exist between different data compression techniques, and how do they impact data integrity and size?
- How can information be securely encrypted or hidden, and what makes encryption methods effective or vulnerable?
- How do abstractions like pixel grids and color codes allow computers to store and manipulate digital images?
- How can we apply programming concepts to creatively process and protect digital information in real-world contexts?

</td>
<td>

representation affects how data is stored, manipulated, and interpreted.

- Compression techniques involve a balance between efficiency and accuracy, with lossy and lossless methods each offering distinct advantages and limitations.
- Cryptographic systems, including steganography, rely on mathematical algorithms and abstraction to protect privacy and ensure secure communication.
- Abstraction enables developers to manage complexity when working with digital data, such as by using functions to manipulate pixels in an image.
- Creative projects like image filters and steganography demonstrate how computational tools and digital information can be used to solve problems and express ideas.

</td>
</tr>
<tr>
<td colspan="2">Vocabulary: abstraction, digital information, number system, number base, decimal, binary, ASCII, pixel, hexadecimal, RGB, data compression, lossless compression, lossy compression, cryptography, cipher, symmetric encryption, public key encryption, steganography</td>
</tr>
<tr>
<td colspan="2">Students will be able to:

- explore and explain abstraction and the different ways that we can represent digital information.
- Represent numbers in different number systems
- Understand how to convert between the decimal and binary system
- Understand the binary system
- Encode various types of information using binary

</td>
</tr>
</table>

- Understand how images can be encoded as data
- Understand how to convert between the hexadecimal and binary system
- Encode colors
- Encode color images as data
- Include images in their programs
- Manipulate the stored pixel data arbitrarily
- have a basic understanding of how data can be compressed.
- understand different types of compressions, and the benefits and drawbacks to each.
- Explain what cryptography is and how it is used
- Analyze an encryption and determine if it is weak or strong
- Explain the difference between an easy and hard problem in computer science
- Explain the difference between algorithms that run in a reasonable amount of time and those that do not run in a reasonable amount of time
- Explain the difference between solvable and unsolvable problems in computer science
- Explain the existence of undecidable problems in computer science
- Explain the connection between cryptography and encoding information digitally
- Create a program that encodes secret information in the binary pixel data of an image

Learning Activities & Differentiation: *(Amistad & Holocaust, Asian American & Pacific Islanders, Climate Change, LGBTQ and People with Disabilities, Diversity, Equity & Inclusion [DEI])*

Learning Activities:

- Interactive exercises converting between binary, decimal, and hexadecimal number systems
- Pixel image programming tasks where students encode and manipulate image data
- Data compression labs exploring lossy and lossless techniques using text and image files
- Cryptography activities involving the design, testing, and cracking of encryption algorithms
- Steganography practice performance task embedding secret messages in pixel data
- Creative image filter projects using digital abstraction and data manipulation

Differentiation Strategies:

- For Advanced Learners: Design multi-layered encryption systems, lead peer debriefs on data ethics, implement advanced compression algorithms
- For Struggling Learners: Use visual binary/hex grids, scaffold encryption templates, provide step-by-step conversion practice between number systems
- For English Language Learners: Support with bilingual coding glossaries, visual coding environments, and sentence starters for explaining concepts
- Universal Design for Learning: Multiple code editor options, student choice in image content and message encoding, flexible project formats including visual, written, or oral presentations

Education Connections:

- Highlight contributions of diverse pioneers in computing and data encryption
- Explore how data representation and accessibility tools (like alt text or screen readers) support individuals with disabilities
- Use climate datasets to explore compression and visual encoding of environmental change
- Discuss ethical implications of surveillance, data storage, and privacy through the lens of historic and modern events (e.g., misuse of data during the Holocaust, current data profiling concerns)
- Address issues of algorithmic bias, equitable data access, and digital inclusion through open discussions and reflective activities

Assessment Evidence

Formative Assessments:

- CodeHS reflection exercises
    - Example Exercise: Guess the Passcode

      Students first imagine they forgot their 4-digit passcode for their phone, and need to guess the correct passcode. They develop a program to guess passcodes for them to speed up the process. Once the correct

passcode has been guessed, the program should print out how many guesses it took to reach the correct one. This activity encourages students to consider security issues which can be expanded to how we create a safer computing culture

- CodeHS Checks for Understanding

Summative Assessments:

- Practice PT: Steganography
- Practice PT: Create an Image Filter

Common Benchmark Assessments:

- Practice PT: Steganography
- Practice PT: Create an Image Filter

Alternative Assessments:

- CodeHS Unit Quiz

Core Instructional & Supplemental Materials

Core Materials:

- CodeHS JavaScript development environment and interactive coding platform
- Built-in JavaScript documentation (DOCS tab) for working with number systems, image data, and functions
- Pixel image editors and color encoding reference tools (e.g., RGB, Hex, Binary charts)
- Interactive debugging and data visualization tools to support image encoding and encryption logic

Supplemental Materials:

- Binary and hexadecimal conversion charts and scaffolding worksheets
- Visual templates for image grids, color encoding, and data compression comparisons
- Peer collaboration rubrics for pair programming during encryption and steganography projects
- Ethical case studies and discussion prompts on data privacy, surveillance, and algorithmic bias
- Assessment rubrics for digital image filters, data encoding projects, and steganography performance tasks

Accommodations and Modifications:
Students with Disabilities, English Language Learners, Students at Risk of Failure, Students with 504s, Gifted & Talented Students

Specific Strategies and Practices that Support Students with Disabilities:

Students receiving Special Education programming have specific goals and objectives, as well as accommodations and modifications outlined within their Individualized Education Plans (IEP) due to an identified disability and/or diagnosis. In addition to exposure to the general education curriculum, the instruction is differentiated based upon the student's needs. The IEP acts as a supplemental curriculum guide inclusive of instructional strategies that support each learner.

- Small group/One to one - Additional time - Space for movement or breaks - Follow a routine/schedule - Rest breaks - Verbal and visual cues regarding directions and staying on task

Specific Strategies and Practices that Support English Language Learners

English language learner (ELL) is a national-origin-minority student who is limited-English-proficient. This term is often preferred over limited-English-proficient (LEP) as it highlights accomplishments rather than deficits.

- Pre-teaching of vocabulary and concepts - Personal Glossary - Extended Time - Simplified/Verbal Instructions - Frequent Breaks - Visual learning, including graphic organizers - Use of cognates to increase comprehension

Students at Risk of Failure

- Home/School Connection - Tiered Level of Supports - Tiered Activities - Small Group Instruction - Work Alone Choice - Group Work Choice

Students with 504 Plans

Teachers are responsible for implementing designated services and strategies identified on a student's 504 Plan.
504 plans are formal plans that schools develop to give kids the support they need and covers any condition that limits daily activities in a major way. Students' accommodations are outlined in their individual 504 plan.

Specific Strategies and Practices that Support Gifted & Talented Students

It is critical that all teachers are able to recognize a high-ability student who may need more depth and complexity in instruction.

-Homogenous Grouping - Differentiated Online Practice - Alternative Assessments

Unit 5

Desired Results

Unit 5 Title: The Internet, Networks, and Cybersecurity

Unit Summary:

This unit explores the structure and design of the internet, and how this design affects the reliability of network communication, the security of data, and personal privacy. Students will learn about the protocols and algorithms used on the internet and the importance of cybersecurity. Students will choose an innovation that was enabled by the Internet and explore the positive and negative impacts of their innovation on society, economy, and culture. Students will develop a computational artifact that illustrates, represents, or explains the innovation's purpose, its function, or its effect, and embed this artifact in their personal portfolio website.

College Board AP CSP Big Ideas covered in Unit 1: CSN, IOC

Learning Standards

NJSLS:

- 8.1.12.NI.1 – Evaluate the scalability and reliability of networks, by describing the relationship between routers, switches, servers, topology, and addressing.
- 8.1.12.NI.2 – Evaluate security measures to address various common security threats.
- 8.1.12.NI.3 – Explain how the needs of users and the sensitivity of data determine the level of security implemented.
- 8.1.12.NI.4 – Explain how decisions on methods to protect data are influenced by whether the data is at rest, in transit, or in use.
- 8.1.12.IC.1 – Evaluate the ways computing impacts personal, ethical, social, economic, and cultural practices.
- 8.1.12.IC.2 – Test and refine computational artifacts to reduce bias and equity deficits.
- 8.1.12.IC.3 – Predict the potential impacts and implications of emerging technologies on larger social, economic,

and political structures, using evidence from credible sources.

- 8.1.12.DA.2 – Describe the trade-offs in how and where data is organized and stored.
- 8.1.12.DA.3 – Translate between decimal numbers and binary numbers.
- 8.1.12.DA.4 – Explain the relationship between binary numbers and the storage and use of data in a computing device.
- 8.1.12.CS.1 – Describe ways in which integrated systems hide underlying implementation details to simplify user experiences.
- 8.1.12.CS.2 – Model interactions between application software, system software, and hardware.

Interdisciplinary Connections

Career Readiness, Life Literacies, & Key Skills *(CLKS)*:

- 9.4.12.DC.1 – Explain the beneficial and harmful effects that intellectual property laws can have on the creation and sharing of content.
- 9.4.12.DC.2 – Compare and contrast international differences in copyright laws and ethics.
- 9.4.12.DC.3 – Evaluate the social and economic implications of privacy in the context of safety, law, or ethics.
- 9.4.12.DC.4 – Explain the privacy concerns related to the collection of data and generation of data through automated processes that may not be evident to users.
- 9.4.12.DC.5 – Debate laws and regulations that impact the development and use of software.
- 9.4.12.DC.7 – Evaluate the influence of digital communities on the nature, content, and responsibilities of careers, and other aspects of society.
- 9.4.12.DC.8 – Explain how increased network connectivity and computing capabilities of everyday objects allow for innovative technological approaches to climate protection.
- 9.4.12.IML.2 – Evaluate digital sources for timeliness, accuracy, perspective, credibility, and relevance.
- 9.4.12.IML.3 – Analyze data using tools and models to make valid and reliable claims, or to determine optimal design solutions.
- 9.4.12.IML.4 – Assess and critique the appropriateness and impact of existing data visualizations for an intended audience.
- 9.4.12.IML.7 – Develop an argument to support a claim regarding a current workplace or societal/ethical issue such as climate change.

- 9.4.12.IML.8 – Evaluate media sources for point of view, bias, and motivations.
- 9.4.12.IML.9 – Analyze the decisions creators make to reveal explicit and implicit messages within information and media.
- 9.4.12.TL.1 – Assess digital tools based on features such as accessibility options, capacities, and utility for accomplishing a specified task.
- 9.4.12.TL.3 – Analyze the effectiveness of the process and quality of collaborative environments.
- 9.4.12.TL.4 – Collaborate in online learning communities or social networks or virtual worlds to analyze and propose a resolution to a real-world problem.
- 9.4.12.CT.1 – Identify problem-solving strategies used in the development of an innovative product or practice.
- 9.4.12.CT.2 – Explain the potential benefits of collaborating to enhance critical thinking and problem solving.

Interdisciplinary Connections:

Science and Engineering Practices:

- Develop and use a model based on evidence to illustrate the relationships between systems or between components of a system.
- Use a model to predict the relationships between systems or between components of a system.
- Analyze data using tools, technologies, and/or models (e.g., computational, mathematical) in order to make valid and reliable scientific claims or determine an optimal design solution.
- Communicate scientific and technical information (e.g. about the process of development and the design and performance of a proposed process or system) in multiple formats (including orally, graphically, textually, and mathematically).
- Create a computational model or simulation of a phenomenon, designed device, process, or system.
- Design, evaluate, and/or refine a solution to a complex real-world problem, based on scientific knowledge, student-generated sources of evidence, prioritized criteria, and tradeoff considerations.
- Evaluate the claims, evidence, and reasoning behind currently accepted explanations or solutions to determine the merits of arguments.

- Use mathematical and/or computational representations of phenomena or design solutions to support explanations.
- Apply scientific reasoning to link evidence to the claims to assess the extent to which the reasoning and data support the explanation or conclusion.
- Construct an oral and written argument or counter-arguments based on data and evidence.
- Evaluate a solution to a complex real-world problem, based on scientific knowledge, student-generated sources of evidence, prioritized criteria, and tradeoff considerations.

Engineering Design Standards:

- HS-ETS1-1 Analyze a major global challenge to specify qualitative and quantitative criteria and constraints for solutions that account for societal needs and wants.
- HS-ETS1-2 Design a solution to a complex real-world problem by breaking it down into smaller, more manageable problems that can be solved through engineering.
- HS-ETS1-3 Evaluate a solution to a complex real-world problem based on prioritized criteria and trade-offs that account for a range of constraints, including cost, safety, reliability, and aesthetics, as well as possible social, cultural, and environmental impacts.
- HS-ETS1-4 Use a computer simulation to model the impact of proposed solutions to a complex real-world problem with numerous criteria and constraints on interactions within and between systems relevant to the problem.

Learning Targets

Unit Essential Questions:	**Unit Enduring Understandings:**
• How does the design and structure of the Internet ensure reliable and efficient communication across global networks? • What are the benefits and risks associated with sharing personal data over the Internet?	• The Internet is a complex, scalable system built on protocols, redundancy, and fault-tolerance, which allows for reliable global communication.

<table>
<tr>
<td>

- How do protocols and internet infrastructure support the flow of information, and what happens when parts of the system fail?
- How has the Internet enabled innovations that impact society, economy, and culture—both positively and negatively?
- What ethical and cybersecurity concerns arise with widespread internet use, and how can individuals and organizations address them responsibly?

</td>
<td>

- Protocols like TCP/IP, HTTP, and DNS govern how information is transferred and routed, ensuring accurate and efficient data delivery.
- The Internet enables transformative innovations, but these innovations can have unintended consequences on individuals, communities, and cultures.
- Users' digital actions leave a footprint, and understanding privacy, authentication, and encryption is critical to protecting personal data.
- Ethical and legal issues—including copyright, digital divide, and cybersecurity threats—must be considered when designing and using internet-based technologies.

</td>
</tr>
<tr>
<td colspan="2">Vocabulary: internet, protocol, network, bitrate, bandwidth, latency, network device, computing device, input device, output device, storage device, computing system, computer network, IP, IPv6, URL, HTTP, DNS, routing, redundancy, packets, TCP, sequential programming, parallel programming, distributed programming, spoofing, DDoS attack, cybersecurity, public key encryption, copyright, public domain</td>
</tr>
<tr>
<td colspan="2">Students will be able to:

- Understand what the internet is
- Understand how the internet works
- Discuss the issue of anonymity
- Understand the legal and ethical concerns surrounding internet censorship
- Discuss and answer questions about the hardware that powers the internet
- Discuss the necessity of internet protocols
- Recognize the hierarchy of elements in an IP address

</td>
</tr>
</table>

- Describe the process that occurs when typing in a URL, from sending a request and response over the Internet to viewing a webpage
- Understand the DNS system and how it works
- Recognize the DNS system as an abstraction
- Explain how computers communicate using routers
- Explain what considerations are made when choosing a route
- Discuss how routers are fault-tolerant because of redundancy
- Explain the packet process and how protocols (TCP/IP and HTTP) are vital to the exchange of information on the Internet
- Explain the HyperText Transfer Protocol
- differentiate between sequential, parallel, and distributed computing
- explain the benefits and challenges of using parallel and distributed computing
- have an understanding of why cybersecurity is necessary, and some practical measures that they can take themselves to improve their security on the internet.
- Analyze the different ways that the Internet impacts their lives by learning about how the Internet contributes to collaboration, communication, etc
- Evaluate whether the Internet has a more positive or negative effect on their community by citing examples from the lesson
- Explain what the digital divide is and articulate their own opinions related to it
- Explain what copyright laws are and why they are important
- Find images they are legally allowed to use in their projects
- Accurately attribute images they find and want to use
- Communicate the beneficial and harmful effects of a computing innovation
- Develop a computational artifact to represent a computing innovation of choice
- Cite relevant and credible sources in their arguments
- Gain practice for the AP Explore Performance Task

Learning Activities & Differentiation: *(Amistad & Holocaust, Asian American & Pacific Islanders, Climate Change, LGBTQ and*

People with Disabilities, Diversity, Equity & Inclusion [DEI])

Learning Activities:

- Simulations demonstrating how data travels across networks using IP, DNS, and HTTP protocols
- Interactive labs visualizing routing, packet loss, redundancy, and fault tolerance
- Activities decoding and encoding data using binary, hexadecimal, and ASCII systems
- Cybersecurity challenges including phishing simulations, password strength analysis, and encryption/decryption tasks
- Group research and presentations on internet-enabled innovations and their impact on society, economy, and culture
- Creative performance tasks developing computational artifacts that explore the benefits and risks of digital technologies
- Case study discussions on digital privacy, surveillance, and the digital divide

Differentiation Strategies:

- For Advanced Learners: Develop complex case studies on cybersecurity breaches and propose layered solutions
- For Struggling Learners: Scaffold internet routing activities with guided worksheets and protocol flowcharts
- For English Language Learners: Use visual aids and diagram-based explanations for technical systems
- Universal Design for Learning: Multiple representation formats for student work (e.g., videos, diagrams, websites, or oral presentations)

Education Connections:

- Discuss the role of encryption during historical events such as WWII and its evolution in modern cybersecurity
- Highlight internet and computing contributions from diverse global leaders, including Asian American and Pacific Islander technologists
- Analyze how internet access and the digital divide impact marginalized communities
- Use environmental datasets to demonstrate the role of internet-connected systems in climate monitoring and awareness

- Explore accessibility technologies and inclusive design through real-world examples (e.g., screen readers, alt text, internet access equity)
- Examine surveillance and data profiling practices historically (e.g., during the Holocaust) and today, and their impact on civil rights

Assessment Evidence

Formative Assessments:

- CodeHS reflection exercises
    - Example Exercise: Packets and Protocols The Story of the Internet

      In their own words, students tell the story of downloading an image from a website on the internet. They tell the story step by step of how their computer finds the relevant server, requests information from the server, and receives it. Students are required to include distinctions between the internet and the World Wide Web.
- CodeHS Checks for Understanding

Summative Assessments:

- Project: The Effects of the Internet

Common Benchmark Assessments:

- Project: The Effects of the Internet

Alternative Assessments:

- CodeHS Unit Quiz

Core Instructional & Supplemental Materials

Core Materials:

- CodeHS Internet & Cybersecurity modules and interactive coding platform

- Built-in JavaScript documentation (DOCS tab) for exploring protocols, addressing systems, and cybersecurity functions
- Internet simulators and packet tracing tools to visualize data flow, redundancy, and fault tolerance
- DNS and IP address lookup tools to support understanding of addressing and routing concepts
- Computational artifact creation tools for illustrating the impact of internet-enabled innovations

Supplemental Materials:

- Binary, hexadecimal, and ASCII conversion charts and scaffolding worksheets
- Visual templates for packet diagrams, routing models, and distributed computing comparisons
- Peer collaboration rubrics for pair programming and group projects on cybersecurity and computing innovations
- Ethical case studies and discussion prompts on digital footprint, surveillance, censorship, and cybersecurity threats
- Assessment rubrics for impact analysis essays, internet innovation artifacts, and performance tasks related to data transmission and security

Accommodations and Modifications: Students with Disabilities, English Language Learners, Students at Risk of Failure, Students with 504s, Gifted & Talented Students

Specific Strategies and Practices that Support Students with Disabilities:

Students receiving Special Education programming have specific goals and objectives, as well as accommodations and modifications outlined within their Individualized Education Plans (IEP) due to an identified disability and/or diagnosis. In addition to exposure to the general education curriculum, the instruction is differentiated based upon the student's needs. The IEP acts as a supplemental curriculum guide inclusive of instructional strategies that support each learner.

- Small group/One to one - Additional time - Review of directions - Student restates information - Space for movement or breaks - Extra visual and verbal cues and prompts - Preferential seating - Follow a routine/schedule - Rest breaks - Verbal and visual cues regarding directions and staying on task - Checklists - Immediate feedback

Specific Strategies and Practices that Support English Language Learners

English language learner (ELL) is a national-origin-minority student who is limited-English-proficient. This term is often preferred over limited-English-proficient (LEP) as it highlights accomplishments rather than deficits.

- Pre-teaching of vocabulary and concepts- Extended Time - Simplified/Verbal Instructions - Teacher modeling - Pairing students with beginning English language skills with students who have more advanced English language skills - Scaffolding - Think-pair-share - Cooperative learning groups

Students at Risk of Failure

- Home/School Connection - Learning Contracts - Homogenous Grouping - Think-Pair-Share by Readiness - Homework Options - Differentiated Online Practice - Leveled Rubrics - Tiered Level of Supports - Small Group Instruction - Work Alone Choice - Group Work Choice

Students with 504 Plans

Teachers are responsible for implementing designated services and strategies identified on a student's 504 Plan.
504 plans are formal plans that schools develop to give kids the support they need and covers any condition that limits daily activities in a major way. Students' accommodations are outlined in their individual 504 plan.

Specific Strategies and Practices that Support Gifted & Talented Students

It is critical that all teachers are able to recognize a high-ability student who may need more depth and complexity in instruction.

- Differentiated Online Practice - Alternative Assessments - Pre-Assess to provide Differentiated Instruction to Accelerate Learning

Unit 6

Desired Results

Unit 6 Title: Data Analysis and Visualization

Unit Summary:

In this unit, students will explore using computational tools to store massive amounts of data, manipulate and visualize data, find patterns in data, and draw conclusions from data. Students will consider how the modern wealth of data collection has impacted society in positive and negative ways. Students will work in teams to investigate a question of personal interest and use public data to present a data-driven insight to their peers. They will develop visualizations to communicate their findings, and embed their visualizations in their portfolio websites.

College Board AP CSP Big Ideas covered in Unit 1: DAT, CRD

Learning Standards

NJSLS:

- 8.1.12.CS.1: Describe ways in which integrated systems hide underlying implementation details to simplify user experiences.
- 8.1.12.CS.2: Model interactions between application software, system software, and hardware.
- 8.1.12.CS.3: Compare the functions of application software, system software, and hardware.
- 8.1.12.IC.1: Evaluate the ways computing impacts personal, ethical, social, economic, and cultural practices.
- 8.1.12.IC.2: Test and refine computational artifacts to reduce bias and equity deficits.
- 8.1.12.IC.3: Predict the potential impacts and implications of emerging technologies on larger social, economic, and political structures, using evidence from credible sources.
- 8.1.12.DA.1: Create interactive data visualizations using software tools to help others better understand real world phenomena, including climate change.
- 8.1.12.DA.2: Describe the trade-offs in how and where data is organized and stored.
- 8.1.12.DA.5: Create data visualizations from large data sets to summarize, communicate, and support different interpretations of real-world phenomena.
- 8.1.12.DA.6: Create and refine computational models to better represent the relationships among different

elements of data collected from a phenomenon or process.

- 8.1.12.AP.4: Design and iteratively develop computational artifacts for practical intent, personal expression, or to address a societal issue.
- 8.1.12.AP.7: Collaboratively design and develop programs and artifacts for broad audiences by incorporating feedback from users.
- 8.1.12.AP.8: Evaluate and refine computational artifacts to make them more usable and accessible.
- 8.2.12.ED.1: Use research to design and create a product or system that addresses a problem and make modifications based on input from potential consumers.
- 8.2.12.ED.4: Design a product or system that addresses a global problem and document decisions made based on research, constraints, trade-offs, and aesthetic and ethical considerations and share this information with an appropriate audience.
- 8.2.12.EC.1: Analyze controversial technological issues and determine the degree to which individuals, businesses, and governments have an ethical role in decisions that are made.
- 8.2.12.EC.3: Synthesize data, analyze trends, and draw conclusions regarding the effect of a technology on the individual, culture, society, and environment and share this information with the appropriate audience.

Interdisciplinary Connections

Career Readiness, Life Literacies, & Key Skills *(CLKS)*:

- 9.1.12.CFR.1: Compare and contrast the role of philanthropy, volunteer service, and charities in community development and quality of life in a variety of cultures.
- 9.1.12.CFR.3: Research companies with corporate governance policies supporting the common good and human rights.
- 9.1.12.CFR.4: Demonstrate an understanding of the interrelationships among attitudes, assumptions, and patterns of behavior regarding money, saving, investing, and work across cultures.
- 9.2.12.CAP.2: Develop college and career readiness skills by participating in opportunities such as structured learning experiences, apprenticeships, and dual enrollment programs.
- 9.2.12.CAP.3: Investigate how continuing education contributes to one's career and personal growth.
- 9.2.12.CAP.5: Assess and modify a personal plan to support current interests and postsecondary plans.
- 9.2.12.CAP.6: Identify transferable skills in career choices and design alternative career plans based on those skills.

- 9.2.12.CAP.7: Use online resources to examine licensing, certification, and credentialing requirements at the local, state, and national levels to maintain compliance with industry requirements in areas of career interest.
- 9.2.12.CAP.14: Analyze and critique various sources of income and available resources (e.g., financial assets, property, and transfer payments) and how they may substitute for earned income.
- 9.4.12.CI.1: Demonstrate the ability to reflect, analyze, and use creative skills and ideas (e.g., 1.1.12prof.CR3a).
- 9.4.12.CI.2: Identify career pathways that highlight personal talents, skills, and abilities (e.g., 1.4.12prof.CR2b, 2.2.12.LF.8).
- 9.4.12.CT.1: Identify problem-solving strategies used in the development of an innovative product or practice (e.g., 1.1.12acc.C1b, 2.2.12.PF.3).
- 9.4.12.CT.2: Explain the potential benefits of collaborating to enhance critical thinking and problem solving (e.g., 1.3E.12profCR3.a).
- 9.4.12.CT.3: Enlist input from a variety of stakeholders (e.g., community members, experts in the field) to design a service learning activity that addresses a local or global issue (e.g., environmental justice).
- 9.4.12.CT.4: Participate in online strategy and planning sessions for course-based, school-based, or other project and determine the strategies that contribute to effective outcomes.
- 9.4.12.DC.1: Explain the beneficial and harmful effects that intellectual property laws can have on the creation and sharing of content (e.g., 6.1.12.CivicsPR.16.a).
- 9.4.12.DC.3: Evaluate the social and economic implications of privacy in the context of safety, law, or ethics (e.g., 6.3.12.HistoryCA.1).
- 9.4.12.DC.4: Explain the privacy concerns related to the collection of data (e.g., cookies) and generation of data through automated processes that may not be evident to users (e.g., 8.1.12.NI.3).
- 9.4.12.DC.6: Select information to post online that positively impacts personal image and future college and career opportunities.
- 9.4.12.DC.7: Evaluate the influence of digital communities on the nature, content and responsibilities of careers, and other aspects of society (e.g., 6.1.12.CivicsPD.16.a).
- 9.4.12.DC.8: Explain how increased network connectivity and computing capabilities of everyday objects allow for innovative technological approaches to climate protection.

- 9.4.12.GCA.1: Collaborate with individuals to analyze a variety of potential solutions to climate change effects and determine why some solutions (e.g., political, economic, cultural) may work better than others.
- 9.4.12.IML.1: Compare search browsers and recognize features that allow for filtering of information.
- 9.4.12.IML.2: Evaluate digital sources for timeliness, accuracy, perspective, credibility of the source, and relevance of information, in media, data, or other resources.
- 9.4.12.IML.3: Analyze data using tools and models to make valid and reliable claims, or to determine optimal design solutions.
- 9.4.12.IML.4: Assess and critique the appropriateness and impact of existing data visualizations for an intended audience.
- 9.4.12.IML.5: Evaluate, synthesize, and apply information on climate change from various sources appropriately.
- 9.4.12.IML.6: Use various types of media to produce and store information on climate change for different purposes and audiences with sensitivity to cultural, gender, and age diversity.
- 9.4.12.IML.7: Develop an argument to support a claim regarding a current workplace or societal/ethical issue such as climate change.
- 9.4.12.IML.8: Evaluate media sources for point of view, bias, and motivations.
- 9.4.12.IML.9: Analyze the decisions creators make to reveal explicit and implicit messages within information and media.
- 9.4.12.TL.1: Assess digital tools based on features such as accessibility options, capacities, and utility for accomplishing a specified task.
- 9.4.12.TL.2: Generate data using formula-based calculations in a spreadsheet and draw conclusions about the data.
- 9.4.12.TL.3: Analyze the effectiveness of the process and quality of collaborative environments.
- 9.4.12.TL.4: Collaborate in online learning communities or social networks or virtual worlds to analyze and propose a resolution to a real-world problem.

Interdisciplinary Connections:

Science and Engineering Practices:

- Develop a model based on evidence to illustrate the relationships between systems or between components of a system.
- Use a model to predict the relationships between systems or between components of a system.
- Analyze data using tools, technologies, and/or models (e.g., computational, mathematical) in order to make valid and reliable scientific claims or determine an optimal design solution.
- Use mathematical and/or computational representations of phenomena or design solutions to support explanations.
- Apply concepts of statistics and probability (including determining function fits to data, slope, intercept, and correlation coefficient for linear fits) to scientific and engineering questions and problems, using digital tools when feasible.
- Communicate scientific and technical information (e.g. about the process of development and the design and performance of a proposed process or system) in multiple formats (including orally, graphically, textually, and mathematically).
- Create a computational model or simulation of a phenomenon, designed device, process, or system.
- Evaluate the validity and reliability of multiple claims that appear in scientific and technical texts or media reports, verifying the data when possible.

Engineering Design Standards:

- HS-ETS1-4 Use a computer simulation to model the impact of proposed solutions to a complex real-world problem with numerous criteria and constraints on interactions within and between systems relevant to the problem.

Learning Targets

Unit Essential Questions:	**Unit Enduring Understandings:**
• How can data be used to identify trends, patterns, and relationships that inform decision-making?	• Data can be filtered, organized, and visualized to reveal patterns and support claims.

• What role do computational tools play in analyzing and visualizing large datasets? • How does metadata influence the way we interpret and use data? • What are the limitations and ethical considerations of using public data sources? • How can we effectively communicate insights from data through visualizations and digital artifacts?	• Computational tools help manage, process, and interpret large and complex data sets efficiently. • Metadata enhances the utility and meaning of data by providing context and descriptive information. • Data analysis can be impacted by limitations such as bias, incompleteness, or misinterpretation. • Clear and ethical communication of data insights is essential for informed decision-making and societal impact.
Vocabulary: data, data visualization, metadata, correlation, data extraction	
Students will be able to: • learn how computers are used to collect, store, manipulate, and visualize data in order to answer questions and gain knowledge of the world. • Explain the importance of visually depicting data to make information easier to use and to understand trends and changes in information • Understand how computers collect and store data • Analyze data interpretation by learning ways in which data can be skewed • How to think meta-cognitively about the data being represented • collaborate to process data and gain knowledge about a question of interest to them, and present their data driven insight to their classmates • Create a personal artifact using computing tools to illustrate a hobby or interest. • Identify and evaluate computing innovations that impact society, the economy, and culture. • Analyze different data types and their role in computing applications. • Investigate data privacy policies and evaluate potential privacy/security risks.	
Learning Activities & Differentiation: *(Amistad & Holocaust, Asian American & Pacific Islanders, Climate Change, LGBTQ and*	

People with Disabilities, Diversity, Equity & Inclusion [DEI])

Learning Activities:

- Scaffolded team projects to investigate personal interest questions using public data sets, including planning, data collection, cleaning, and analysis
- Guided workshops on creating interactive data visualizations using computational tools to communicate findings effectively
- Collaborative peer review sessions to evaluate visualizations for clarity, bias reduction, accessibility, and cultural sensitivity
- Discussions on ethical implications of data collection, privacy concerns, and potential social impacts of data-driven decisions
- Case studies examining how data analysis can reveal disparities and promote social justice in diverse communities
- Opportunities to embed visualizations into portfolio websites with attention to inclusive design and accessibility features
- Exploration of how network connectivity, AI, and emerging tech influence data use in climate change and societal applications
- Mini-challenges to refine computational artifacts based on feedback, focusing on usability, inclusivity, and ethical considerations

Differentiation Strategies:

- For Advanced Learners: Encourage use of advanced data modeling, multi-layered visualizations, or integration of real-time data streams
- For Struggling Learners: Provide structured templates for data analysis steps, guided tutorials on visualization tools, and vocabulary support
- For English Language Learners: Offer bilingual glossaries, sentence frames for data presentation, and scaffolded writing supports for reflections
- Universal Design for Learning: Allow demonstrations of understanding via oral presentations, video walkthroughs

of data projects, or annotated visual portfolios

Education Connections:

- Analyze historical and modern examples of data use and misuse in marginalized communities, including surveillance and privacy during the Holocaust and present-day issues
- Highlight contributions and challenges faced by Asian American and Pacific Islander communities in data-driven tech fields and society
- Investigate the digital divide and how inequities in data access affect marginalized groups, encouraging projects that promote data equity
- Explore data visualization and computational approaches to climate change, encouraging student-led solutions and awareness campaigns
- Discuss the role of accessible technology and data tools for people with disabilities, including ethical considerations in data collection and usage
- Evaluate the impact of algorithmic bias and representation in data science on LGBTQ+ populations and other underrepresented groups, with focus on ethical artifact design

Assessment Evidence

Formative Assessments:

- CodeHS reflection exercises
    - Example Exercise: Importance of Metadata

      Students consider how metadata can increase the effective use of data or data sets by providing additional information. They consider the importance of metadata and reflect on why metadata is important for a data set, how metadata help in finding specific data, and what metadata should reveal about the data
- CodeHS Checks for Understanding

Summative Assessments:

- Project: Present a Data-Driven Insight

- Project: The Impacts of Computing

Common Benchmark Assessments:

- Project: Present a Data-Driven Insight
- Project: The Impacts of Computing

Alternative Assessments:

- CodeHS Unit Quiz

Core Instructional & Supplemental Materials

Core Materials:

- CodeHS Data and Visualization modules and interactive coding platform for exploring data collection, filtering, analysis, and visual representation
- Built-in JavaScript documentation (DOCS tab) for referencing data structures, visualization tools, and data manipulation functions
- Visualization tools and spreadsheet applications (e.g., chart generators, Google Sheets) to create bar graphs, scatter plots, and pie charts from real-world data
- Public dataset repositories and data search tools (e.g., Data.gov, Kaggle) to support student-led exploration and project development
- Computational artifact creation tools for embedding charts and written insights into student portfolio websites

Supplemental Materials:

- Visual templates for creating effective charts and dashboards including guides for selecting appropriate types of visualizations
- Peer collaboration rubrics for data-driven group investigations and presentation projects
- Ethical case studies and discussion prompts related to data privacy, bias, surveillance, and the societal impact of data collection
- Assessment rubrics for evaluating data visualizations, insight presentations, and portfolio-embedded computational artifacts

Accommodations and Modifications:
Students with Disabilities, English Language Learners,
Students at Risk of Failure, Students with 504s, Gifted & Talented Students

Specific Strategies and Practices that Support Students with Disabilities:

Students receiving Special Education programming have specific goals and objectives, as well as accommodations and modifications outlined within their Individualized Education Plans (IEP) due to an identified disability and/or diagnosis. In addition to exposure to the general education curriculum, the instruction is differentiated based upon the student's needs. The IEP acts as a supplemental curriculum guide inclusive of instructional strategies that support each learner.

- Space for movement or breaks - Extra visual and verbal cues and prompts - Preferential seating - Follow a routine/schedule - Rest breaks - Verbal and visual cues regarding directions and staying on task - Checklists - Immediate feedback

Specific Strategies and Practices that Support English Language Learners

English language learner (ELL) is a national-origin-minority student who is limited-English-proficient. This term is often preferred over limited-English-proficient (LEP) as it highlights accomplishments rather than deficits.

- Pre-teaching of vocabulary and concepts - Personal Glossary - Extended Time - Simplified/Verbal Instructions - Frequent Breaks - Think-pair-share - Cooperative learning groups

Students at Risk of Failure

- Home/School Connection - Alternative Assessments - Learning Contracts - Homogenous Grouping - Supplemental Instruction - Think-Pair-Share by Readiness

Students with 504 Plans

Teachers are responsible for implementing designated services and strategies identified on a student's 504 Plan.
504 plans are formal plans that schools develop to give kids the support they need and covers any condition that limits daily activities in a major way. Students' accommodations are outlined in their individual 504 plan.

Specific Strategies and Practices that Support Gifted & Talented Students

It is critical that all teachers are able to recognize a high-ability student who may need more depth and complexity in instruction.

-Homogenous Grouping - Small Group In-Depth Instruction - Differentiated Online Practice

Unit 7

Desired Results

Unit 7 Title: Create Performance Task and Exam Review

Unit Summary:

This time is set aside for students to prepare for the MCQs and create their AP Create Performance Task. Students will be given the chance to review course content and practice the skills necessary to complete the Create Performance Task. The Create PT will be administered over 9 hours of class time.

College Board AP CSP Big Ideas covered in Unit 1: AAP, CRD, IOC

Learning Standards

NJSLS:

- 8.1.12.IC.1: Evaluate the ways computing impacts personal, ethical, social, economic, and cultural practices.
- 8.1.12.IC.2: Test and refine computational artifacts to reduce bias and equity deficits.
- 8.1.12.IC.3: Predict the potential impacts and implications of emerging technologies on larger social, economic, and political structures, using evidence from credible sources.
- 8.1.12.AP.1: Design algorithms to solve computational problems using a combination of original and existing algorithms.
- 8.1.12.AP.3: Select and combine control structures for a specific application based upon performance and readability, and identify trade-offs to justify the choice.
- 8.1.12.AP.4: Design and iteratively develop computational artifacts for practical intent, personal expression, or to address a societal issue.

- 8.1.12.AP.5: Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects.
- 8.1.12.AP.6: Create artifacts by using procedures within a program, combinations of data and procedures, or independent but interrelated programs.
- 8.1.12.AP.7: Collaboratively design and develop programs and artifacts for broad audiences by incorporating feedback from users.
- 8.1.12.AP.8: Evaluate and refine computational artifacts to make them more usable and accessible.
- 8.1.12.AP.9: Collaboratively document and present design decisions in the development of complex programs.
- 8.1.12.CS.4: Develop guidelines that convey systematic troubleshooting strategies that others can use to identify and fix errors.
- 8.2.12.ED.1: Use research to design and create a product or system that addresses a problem and make modifications based on input from potential consumers.
- 8.2.12.ED.4: Design a product or system that addresses a global problem and document decisions made based on research, constraints, trade-offs, and aesthetic and ethical considerations and share this information with an appropriate audience.
- 8.2.12.ED.5: Evaluate the effectiveness of a product or system based on factors that are related to its requirements, specifications, and constraints.
- 8.2.12.EC.1: Analyze controversial technological issues and determine the degree to which individuals, businesses, and governments have an ethical role in decisions that are made.
- 8.2.12.EC.3: Synthesize data, analyze trends, and draw conclusions regarding the effect of a technology on the individual, culture, society, and environment and share this information with the appropriate audience.

Interdisciplinary Connections

Career Readiness, Life Literacies, & Key Skills *(CLKS)*:

- 9.4.12.CI.1: Demonstrate the ability to reflect, analyze, and use creative skills and ideas.
- 9.4.12.CI.2: Identify career pathways that highlight personal talents, skills, and abilities.
- 9.4.12.CT.1: Identify problem-solving strategies used in the development of an innovative product or practice.
- 9.4.12.CT.2: Explain the potential benefits of collaborating to enhance critical thinking and problem solving.

- 9.4.12.CT.4: Participate in online strategy and planning sessions for course-based, school-based, or other projects and determine the strategies that contribute to effective outcomes.
- 9.4.12.DC.1: Explain the beneficial and harmful effects that intellectual property laws can have on the creation and sharing of content.
- 9.4.12.DC.3: Evaluate the social and economic implications of privacy in the context of safety, law, or ethics.
- 9.4.12.DC.5: Debate laws and regulations that impact the development and use of software.
- 9.4.12.DC.6: Select information to post online that positively impacts personal image and future college and career opportunities.
- 9.4.12.IML.2: Evaluate digital sources for timeliness, accuracy, perspective, credibility of the source, and relevance.
- 9.4.12.IML.3: Analyze data using tools and models to make valid and reliable claims.
- 9.4.12.IML.7: Develop an argument to support a claim regarding a current workplace or societal/ethical issue.
- 9.4.12.TL.1: Assess digital tools based on features such as accessibility options, capacities, and utility for accomplishing a specified task.
- 9.4.12.TL.3: Analyze the effectiveness of the process and quality of collaborative environments.

Interdisciplinary Connections:

Science and Engineering Practices:

- Develop and use a model based on evidence to illustrate the relationships between systems or between components of a system.
- Create a computational model or simulation of a phenomenon, designed device, process, or system.
- Design, evaluate, and/or refine a solution to a complex real-world problem, based on scientific knowledge, student-generated sources of evidence, prioritized criteria, and tradeoff considerations.
- Evaluate the claims, evidence, and reasoning behind currently accepted explanations or solutions to determine the merits of arguments.
- Apply scientific reasoning to link evidence to the claims to assess the extent to which the reasoning and data support the explanation or conclusion.
- Construct an oral and written argument or counter-arguments based on data and evidence.

- Design or refine a solution to a complex real-world problem, based on scientific knowledge, student-generated sources of evidence, prioritized criteria, and tradeoff considerations.
- Evaluate a solution to a complex real-world problem, based on scientific knowledge, student-generated sources of evidence, prioritized criteria, and tradeoff considerations.

Engineering Design Standards:

- HS-ETS1-2 Design a solution to a complex real-world problem by breaking it down into smaller, more manageable problems that can be solved through engineering.
- HS-ETS1-3 Evaluate a solution to a complex real-world problem based on prioritized criteria and trade-offs that account for a range of constraints, including cost, safety, reliability, and aesthetics, as well as possible social, cultural, and environmental impacts.
- HS-ETS1-4 Use a computer simulation to model the impact of proposed solutions to a complex real-world problem with numerous criteria and constraints on interactions within and between systems relevant to the problem.

Learning Targets

Unit Essential Questions:

- How do computing innovations impact the way people think, work, and interact?
- What steps are involved in designing, testing, and refining a program to solve a real-world problem?
- How can programmers effectively collaborate and document their work during the development process?
- Why is it important to consider privacy, security, and ethical implications when working with data and computing innovations?

Unit Enduring Understandings:

- Computing innovations can have both positive and negative effects on society, individuals, and the economy.
- Developing a successful program requires iterative development, careful planning, testing, debugging, and documentation.
- Collaboration and clear communication are key to successful software development and innovation.

• What strategies can help students successfully prepare for the AP CSP multiple choice exam?	• Understanding the ethical and legal responsibilities related to computing is essential when dealing with data and innovation. • Mastery of computational thinking practices and course content improves confidence and performance on AP-style assessments.
Vocabulary: sequencing, selection, iteration, list/array, parameter, conditions, test cases, procedure, input, output, functionality, user, algorithm, arguments, data, collection type, purpose	
Students will be able to: • Identify and explain components A, B, and C of the Create Performance Task in AP Computer Science Principles. • Analyze task requirements and generate thoughtful questions to clarify doubts and deepen understanding. • Actively engage in group discussions, share questions, and collaborate with peers to gain a comprehensive understanding of the task and exam. • Understand and apply the guidelines for the Create Performance Task. • Comprehend the Academic Integrity and Plagiarism Policy, incorporating external ideas ethically and providing proper attribution. • Develop effective strategies for preparation and ethical navigation of the Create Performance Task. • Explain the format of the AP Computer Science Principles exam • Identify course topics that they have a strong understanding of • Identify course topics that they need to review	
Learning Activities & Differentiation: *(Amistad & Holocaust, Asian American & Pacific Islanders, Climate Change, LGBTQ and People with Disabilities, Diversity, Equity & Inclusion [DEI])* **Learning Activities:** • Scaffolded project development time for the AP Create Performance Task, including brainstorming, design	

planning, iterative coding, and artifact reflection
- Peer feedback rounds focusing on usability, accessibility, and inclusivity of computational artifacts
- Code review sessions where students identify and fix bugs, reflect on trade-offs, and improve algorithm design
- Discussions on ethical implications of computing innovations and representation in technology fields
- Workshops on crafting written responses (purpose, function, development process) with attention to diverse perspectives
- Case studies exploring how computing innovations affect marginalized communities and global access
- Optional mini-challenges that allow students to address real-world issues such as climate change, accessibility, or digital bias in their Create PT artifact

Differentiation Strategies:
- For Advanced Learners: Encourage use of APIs, data visualizations, or complex control structures in Create PT artifacts
- For Struggling Learners: Provide exemplars and structured planning templates to organize ideas and manage tasks
- For English Language Learners: Offer sentence starters, graphic organizers, and vocabulary support for written responses
- Universal Design for Learning: Allow students to document their process through videos, annotated code walkthroughs, or oral explanations in addition to written work

Education Connections:
- Examine how computing innovations have both supported and challenged civil rights throughout history, including surveillance practices during the Holocaust and their modern parallels in data collection
- Highlight contributions to computing and software development by Asian American and Pacific Islander innovators and their influence on modern technologies
- Analyze how limited access to technology and the internet (digital divide) impacts marginalized communities and how student projects can promote digital equity

- Encourage students to create projects that address climate change using environmental datasets or simulations to raise awareness through computing
- Explore accessibility features in computing (e.g., screen readers, closed captions, keyboard navigation) and challenge students to include inclusive design in their projects
- Discuss how algorithmic bias and lack of representation in tech can influence outcomes for LGBTQ+ individuals and people with disabilities, and how students can address this in their artifact design

Assessment Evidence

Formative Assessments:

- CodeHS reflection exercises
- CodeHS Checks for Understanding
- Create Task Peer Reviews

Summative Assessments:

- Create Performance Task
- Practice AP Exam

Common Benchmark Assessments:

- Create Performance Task
- Practice AP Exam

Alternative Assessments:

- Additional AP Practice Exams

Core Instructional & Supplemental Materials

Core Materials:

- CodeHS Create Performance Task module and interactive coding platform for guided program development, testing, and documentation
- Built-in JavaScript documentation (DOCS tab) for referencing language syntax, functions, and debugging strategies
- Collaborative development environments (e.g., CodeHS pair programming features or shared IDEs) to support teamwork and peer review
- Student planning templates and development logs for managing incremental program design and iterative testing
- AP Create Performance Task guidelines and exemplars for understanding College Board expectations and documentation requirements
- AP Classroom and CodeHS practice exams and MCQs to reinforce course content and simulate real test conditions

Supplemental Materials:

- Rubrics for evaluating Create PT components, including program purpose, algorithm implementation, abstraction, and data usage
- Reflection prompts and peer review forms to support student metacognition and collaborative feedback
- Debugging checklists and problem-solving guides for resolving issues during development
- Ethical scenarios and discussion prompts related to computing innovations, privacy, and data usage in student-created programs
- Visual planning tools such as flowcharts and pseudocode organizers for program structure and logic
- Review packets and content summaries aligned to AP CSP Big Ideas and Learning Objectives

Accommodations and Modifications: Students with Disabilities, English Language Learners, Students at Risk of Failure, Students with 504s, Gifted & Talented Students

Specific Strategies and Practices that Support Students with Disabilities:

Students receiving Special Education programming have specific goals and objectives, as well as accommodations and modifications outlined within their Individualized Education Plans (IEP) due to an identified disability and/or diagnosis. In addition to exposure to the general education curriculum, the instruction is differentiated based upon the student's needs. The IEP acts as a supplemental curriculum guide inclusive of instructional strategies that support each learner.

- Small group/One to one - Additional time - Review of directions - Preferential seating - Follow a routine/schedule - Rest breaks - Verbal and visual cues regarding directions and staying on task - Checklists

Specific Strategies and Practices that Support English Language Learners

English language learner (ELL) is a national-origin-minority student who is limited-English-proficient. This term is often preferred over limited-English-proficient (LEP) as it highlights accomplishments rather than deficits.

- Extended Time - Simplified/Verbal Instructions - Frequent Breaks - Visual learning, including graphic organizers - Pairing students with beginning English language skills with students who have more advanced English language skills - Scaffolding

Students at Risk of Failure

- Alternative Assessments - Learning Contracts - Homogenous Grouping - Supplemental Instruction - Think-Pair-Share by Readiness - Differentiated Online Practice - Leveled Rubrics - Tiered Activities - Work Alone Choice - Group Work Choice

Students with 504 Plans

Teachers are responsible for implementing designated services and strategies identified on a student's 504 Plan.
504 plans are formal plans that schools develop to give kids the support they need and covers any condition that limits daily activities in a major way. Students' accommodations are outlined in their individual 504 plan.

Specific Strategies and Practices that Support Gifted & Talented Students

It is critical that all teachers are able to recognize a high-ability student who may need more depth and complexity in instruction.

-Homogenous Grouping - Small Group In-Depth Instruction - Differentiated Online Practice - Alternative Assessments - Pre-Assess to provide Differentiated Instruction to Accelerate Learning

Unit 8

Desired Results

Unit 8 Title: Creative Development Final Projects

Unit Summary:

In this unit, students will brainstorm their own final project, discuss their ideas with their peers, scope their project to fit within the time constraints of the class, plan out milestones for incremental development, and create their own final product from scratch. This project allows students to think creatively about the applications of the concepts covered in the course, and create something of personal value.

College Board AP CSP Big Ideas covered in Unit 1: CRD, AAP

Learning Standards

NJSLS:

- 8.1.12.CS.1: Describe ways in which integrated systems hide underlying implementation details to simplify user experiences.
- 8.1.12.CS.4: Develop guidelines that convey systematic troubleshooting strategies that others can use to identify and fix errors.
- 8.1.12.IC.1: Evaluate the ways computing impacts personal, ethical, social, economic, and cultural practices.
- 8.1.12.IC.2: Test and refine computational artifacts to reduce bias and equity deficits.
- 8.1.12.IC.3: Predict the potential impacts and implications of emerging technologies on larger social, economic, and political structures, using evidence from credible sources.
- 8.1.12.AP.4: Design and iteratively develop computational artifacts for practical intent, personal expression, or to address a societal issue.
- 8.1.12.AP.7: Collaboratively design and develop programs and artifacts for broad audiences by incorporating feedback from users.
- 8.1.12.AP.8: Evaluate and refine computational artifacts to make them more usable and accessible.

- 8.2.12.ED.1: Use research to design and create a product or system that addresses a problem and make modifications based on input from potential consumers.
- 8.2.12.ED.4: Design a product or system that addresses a global problem and document decisions made based on research, constraints, trade-offs, and aesthetic and ethical considerations and share this information with an appropriate audience.
- 8.2.12.ITH.1: Analyze a product to determine the impact that economic, political, social, and/or cultural factors have had on its design, including its design constraints.
- 8.2.12.EC.1: Analyze controversial technological issues and determine the degree to which individuals, businesses, and governments have an ethical role in decisions that are made.
- 8.2.12.EC.3: Synthesize data, analyze trends, and draw conclusions regarding the effect of a technology on the individual, culture, society, and environment and share this information with the appropriate audience.

Interdisciplinary Connections

Career Readiness, Life Literacies, & Key Skills *(CLKS)*:

- 9.4.12.CI.1: Demonstrate the ability to reflect, analyze, and use creative skills and ideas (e.g., 1.1.12prof.CR3a).
- 9.4.12.CI.2: Identify career pathways that highlight personal talents, skills, and abilities (e.g., 1.4.12prof.CR2b, 2.2.12.LF.8).
- 9.4.12.CI.3: Investigate new challenges and opportunities for personal growth, advancement, and transition (e.g., 2.1.12.PGD.1).
- 9.4.12.CT.1: Identify problem-solving strategies used in the development of an innovative product or practice (e.g., 1.1.12acc.C1b, 2.2.12.PF.3).
- 9.4.12.CT.2: Explain the potential benefits of collaborating to enhance critical thinking and problem solving (e.g., 1.3E.12profCR3.a).
- 9.4.12.CT.3: Enlist input from a variety of stakeholders (e.g., community members, experts in the field) to design a service learning activity that addresses a local or global issue (e.g., environmental justice).
- 9.4.12.CT.4: Participate in online strategy and planning sessions for course-based, school-based, or other project and determine the strategies that contribute to effective outcomes.
- 9.4.12.DC.1: Explain the beneficial and harmful effects that intellectual property laws can have on the creation and sharing of content (e.g., 6.1.12.CivicsPR.16.a).

- 9.4.12.DC.3: Evaluate the social and economic implications of privacy in the context of safety, law, or ethics (e.g., 6.3.12.HistoryCA.1).
- 9.4.12.DC.4: Explain the privacy concerns related to the collection of data (e.g., cookies) and generation of data through automated processes that may not be evident to users (e.g., 8.1.12.NI.3).
- 9.4.12.DC.5: Debate laws and regulations that impact the development and use of software.
- 9.4.12.DC.6: Select information to post online that positively impacts personal image and future college and career opportunities.
- 9.4.12.DC.7: Evaluate the influence of digital communities on the nature, content and responsibilities of careers, and other aspects of society (e.g., 6.1.12.CivicsPD.16.a).
- 9.4.12.DC.8: Explain how increased network connectivity and computing capabilities of everyday objects allow for innovative technological approaches to climate protection.
- 9.4.12.GCA.1: Collaborate with individuals to analyze a variety of potential solutions to climate change effects and determine why some solutions (e.g., political, economic, cultural) may work better than others (e.g., SL.11-12.1., HS-ETS1-1, HS-ETS1-2, HS-ETS1-4, 6.3.12.GeoGI.1, 7.1.IH.IPERS.6, 7.1.IL.IPERS.7, 8.2.12.ETW.3).
- 9.4.12.IML.2: Evaluate digital sources for timeliness, accuracy, perspective, credibility of the source, and relevance of information, in media, data, or other resources (e.g., NJSLSA.W8, Social Studies Practice: Gathering and Evaluating Sources).
- 9.4.12.IML.3: Analyze data using tools and models to make valid and reliable claims, or to determine optimal design solutions (e.g., S-ID.B.6a., 8.1.12.DA.5, 7.1.IH.IPRET.8).
- 9.4.12.IML.4: Assess and critique the appropriateness and impact of existing data visualizations for an intended audience (e.g., S-ID.B.6b, HS-LS2-4).
- 9.4.12.IML.5: Evaluate, synthesize, and apply information on climate change from various sources appropriately (e.g., 2.1.12.CHSS.6, S.IC.B.4, S.IC.B.6, 8.1.12.DA.1, 6.1.12.GeoHE.14.a, 7.1.AL.PRSNT.2).
- 9.4.12.IML.6: Use various types of media to produce and store information on climate change for different purposes and audiences with sensitivity to cultural, gender, and age diversity (e.g., NJSLSA.SL5).
- 9.4.12.IML.7: Develop an argument to support a claim regarding a current workplace or societal/ethical issue such as climate change (e.g., NJSLSA.W1, 7.1.AL.PRSNT.4).
- 9.4.12.IML.8: Evaluate media sources for point of view, bias, and motivations (e.g., NJSLSA.R6, 7.1.AL.IPRET.6).

- 9.4.12.IML.9: Analyze the decisions creators make to reveal explicit and implicit messages within information and media (e.g., 1.5.12acc.C2a, 7.1.IL.IPRET.4).
- 9.4.12.TL.1: Assess digital tools based on features such as accessibility options, capacities, and utility for accomplishing a specified task (e.g., W.11-12.6.).
- 9.4.12.TL.3: Analyze the effectiveness of the process and quality of collaborative environments.
- 9.4.12.TL.4: Collaborate in online learning communities or social networks or virtual worlds to analyze and propose a resolution to a real-world problem (e.g., 7.1.AL.IPERS.6).
- 9.2.12.CAP.2: Develop college and career readiness skills by participating in opportunities such as structured learning experiences, apprenticeships, and dual enrollment programs.
- 9.2.12.CAP.4: Evaluate different careers and develop various plans (e.g., costs of public, private, training schools) and timetables for achieving them, including educational/training requirements, costs, loans, and debt repayment.
- 9.2.12.CAP.6: Identify transferable skills in career choices and design alternative career plans based on those skills.
- 9.2.12.CAP.14: Analyze and critique various sources of income and available resources (e.g., financial assets, property, and transfer payments) and how they may substitute for earned income.

Interdisciplinary Connections:

Science and Engineering Practices:

- Develop and use a model based on evidence to illustrate the relationships between systems or between components of a system.
- Create a computational model or simulation of a phenomenon, designed device, process, or system.
- Design, evaluate, and/or refine a solution to a complex real-world problem, based on scientific knowledge, student-generated sources of evidence, prioritized criteria, and tradeoff considerations.
- Evaluate the claims, evidence, and reasoning behind currently accepted explanations or solutions to determine the merits of arguments.

- Communicate technical information or ideas (e.g. about phenomena and/or the process of development and the design and performance of a proposed process or system) in multiple formats (including orally, graphically, textually, and mathematically).
- Use mathematical and/or computational representations of phenomena or design solutions to support explanations.
- Construct an oral and written argument or counter-arguments based on data and evidence.
- Design or refine a solution to a complex real-world problem, based on scientific knowledge, student-generated sources of evidence, prioritized criteria, and tradeoff considerations.
- Evaluate a solution to a complex real-world problem, based on scientific knowledge, student-generated sources of evidence, prioritized criteria, and tradeoff considerations.

Engineering Design Standards:

- HS-ETS1-1 Analyze a major global challenge to specify qualitative and quantitative criteria and constraints for solutions that account for societal needs and wants.
- HS-ETS1-2 Design a solution to a complex real-world problem by breaking it down into smaller, more manageable problems that can be solved through engineering.
- HS-ETS1-3 Evaluate a solution to a complex real-world problem based on prioritized criteria and trade-offs that account for a range of constraints, including cost, safety, reliability, and aesthetics, as well as possible social, cultural, and environmental impacts.
- HS-ETS1-4 Use a computer simulation to model the impact of proposed solutions to a complex real-world problem with numerous criteria and constraints on interactions within and between systems relevant to the problem.

Learning Targets	
Unit Essential Questions: • What elements contribute to an engaging and accessible user interface?	**Unit Enduring Understandings:**

• How does the iterative design process improve the quality and usability of a product? • Why is user testing critical in the development of effective interfaces? • How do diverse perspectives and human biases influence design decisions? • What ethical and legal considerations must be addressed in user interface design?	• Effective user interfaces balance aesthetics, functionality, and accessibility to create engaging experiences for diverse users. • Iterative development—through prototyping, testing, and refining—is essential to designing user-centered products that meet real needs. • User testing provides invaluable feedback that uncovers usability issues and guides improvements. • Design decisions are influenced by diverse user perspectives and potential human biases, requiring awareness and inclusivity to ensure fairness and accessibility. • Ethical and legal responsibilities, including privacy, accessibility laws, and intellectual property, shape the development and deployment of digital interfaces.
Vocabulary: design thinking, prototype, test	
Students will be able to: • Define design thinking • Name the steps in the design cycle • Explain what prototyping is and why it is an important part of the design process • Describe why testing is an important part of the design process • Explore good and bad testing practices in order to receive helpful feedback on their final project ideas • Connect all they've learned to create a final website • Work in pairs or groups to create a product	

Learning Activities & Differentiation: *(Amistad & Holocaust, Asian American & Pacific Islanders, Climate Change, LGBTQ and People with Disabilities, Diversity, Equity & Inclusion [DEI])*

Learning Activities:

- Scaffolded project development time guiding students through brainstorming, iterative prototyping, user testing, and finalizing their creative web-based UI projects.
- Peer feedback sessions centered on usability, accessibility, inclusivity, and ethical considerations in user interface design.
- User interface scavenger hunts analyzing websites/apps with strong and weak UI elements, emphasizing accessibility and diverse user needs.
- Collaborative workshops on applying design thinking principles while incorporating diverse perspectives and minimizing human biases.
- Discussions and reflections on the ethical and legal implications of UI design, including privacy, accessibility laws, and cultural representation.
- Case studies highlighting how computing and design innovations impact marginalized communities and promote equity in technology access.

Differentiation Strategies:

- For Advanced Learners: Encourage exploring complex UI design tools, integrating accessibility APIs, and incorporating advanced prototyping or animation techniques.
- For Struggling Learners: Provide step-by-step design templates, scaffolded prototyping activities, and visual guides for UI components and user testing.
- For English Language Learners: Offer vocabulary support related to UI design and accessibility, sentence starters for written reflections, and visual organizers for design planning.
- Universal Design for Learning: Allow students to demonstrate their design process through video walkthroughs, interactive prototypes, oral presentations, or annotated sketches in addition to written work.

Education Connections:

- Explore how historical surveillance practices (e.g., during the Holocaust) relate to modern digital privacy concerns and ethical design in user interfaces.
- Highlight contributions from Asian American and Pacific Islander technologists in user experience design and computing innovations.
- Analyze the digital divide's impact on marginalized communities and design solutions promoting equitable access through thoughtful UI design.
- Encourage projects that raise awareness about climate change through interactive data visualizations or accessible environmental web apps.
- Discuss the importance of accessibility features (screen readers, keyboard navigation, color contrast) and challenge students to incorporate inclusive design principles.
- Examine how algorithmic bias and lack of representation in tech affect LGBTQ+ individuals and people with disabilities, and how inclusive UI design can mitigate these issues.

Assessment Evidence

Formative Assessments:

- CodeHS reflection exercises
    - Example Exercise: User Interface Scavenger Hunt

      Students search for 2 websites or apps, one with a good UI and one with a not-so-good UI. They learn to discriminate features of solid UI design in terms of accessibility and more before moving onto prototyping their creative project for the unit.
- CodeHS Checks for Understanding

Summative Assessments:

- Final Project

Common Benchmark Assessments:

- Final Project

Alternative Assessments:

- AP Style MC Exam

Core Instructional & Supplemental Materials

Core Materials:

- Interactive UI/UX design platforms and prototyping tools (e.g., Figma, Adobe XD, or CodeHS Create modules adapted for UI design) for rapid prototyping, iteration, and user testing.
- Design thinking and user interface guidelines documentation, including accessibility standards (WCAG) and usability best practices.
- Collaborative development environments (shared design workspaces, version control platforms, pair programming tools) for teamwork, peer feedback, and collaborative iteration.
- Student planning templates and iterative development logs to scaffold project scope, milestone planning, prototype testing, and design refinements.
- Exemplars and rubrics focused on user-centered design, accessibility, and ethical considerations to clarify expectations and evaluate final projects.
- Visual planning tools such as wireframes, flowcharts, and storyboards to organize UI structure and interaction logic.

Supplemental Materials:

- Resources on ethical, legal, and inclusive design practices, including privacy, bias mitigation, and representation in user interfaces.
- Reflection prompts and peer review forms emphasizing usability, inclusivity, and diverse user needs.
- Supplemental tutorials and content summaries on UI/UX principles, accessibility, human-computer interaction, and iterative design aligned with unit goals.

Accommodations and Modifications:
Students with Disabilities, English Language Learners,

Students at Risk of Failure, Students with 504s, Gifted & Talented Students

Specific Strategies and Practices that Support Students with Disabilities:

Students receiving Special Education programming have specific goals and objectives, as well as accommodations and modifications outlined within their Individualized Education Plans (IEP) due to an identified disability and/or diagnosis. In addition to exposure to the general education curriculum, the instruction is differentiated based upon the student's needs. The IEP acts as a supplemental curriculum guide inclusive of instructional strategies that support each learner.

- Small group/One to one - Additional time - Review of directions - Student restates information - Rest breaks - Verbal and visual cues regarding directions and staying on task - Checklists - Immediate feedback

Specific Strategies and Practices that Support English Language Learners

English language learner (ELL) is a national-origin-minority student who is limited-English-proficient. This term is often preferred over limited-English-proficient (LEP) as it highlights accomplishments rather than deficits.

- Pre-teaching of vocabulary and concepts - Personal Glossary - Extended Time - Simplified/Verbal Instructions - Frequent Breaks - Visual learning, including graphic organizers

Students at Risk of Failure

- Home/School Connection - Alternative Assessments - Learning Contracts - Homogenous Grouping - Supplemental Instruction - Think-Pair-Share by Readiness

Students with 504 Plans

Teachers are responsible for implementing designated services and strategies identified on a student's 504 Plan.
504 plans are formal plans that schools develop to give kids the support they need and covers any condition that limits daily activities in a major way. Students' accommodations are outlined in their individual 504 plan.

Specific Strategies and Practices that Support Gifted & Talented Students

It is critical that all teachers are able to recognize a high-ability student who may need more depth and complexity in instruction.

-Homogenous Grouping - Small Group In-Depth Instruction - Alternative Assessments