Rumson-Fair Haven Regional High School

AP Computer Science Principles Curriculum

Staff Writers: Jodi Foster & Justin Langlois
Supervisor: Christopher Alworth
Approved: 2020

Section I: Course Description

AP Computer Science Principles is a course in logic. It is a college-level introductory course in computer science. This course is NOT considered a Core Math course for college. This course is recommended as a prerequisite to AP Computer Science. "The AP Computer Science Principles course will introduce you to the essential ideas of computer science and show how computing and technology can influence the world around you. You will creatively address real-world issues and concerns while using the same processes and tools as artists, writers, computer scientists, and engineers to bring ideas to life (College Board)." The overarching goal of this course will be to prepare a student to take *The College Board's Advanced Placement Examination in Computer Science Principles*.

Section II: NJSLS: New Jersey Student Learning Standards/Learning Objectives

1. **2020 New Jersey Student Learning Standards – Computer Science and Design Thinking:** **https://www.nj.gov/education/cccs/2020/2020%20NJSLS-CSDT.pdf**
    - "The 'Intent and Spirit of the Computer Science and Design Thinking Standards' is to focus on deep understanding of concepts that enable students to think critically and systematically about leveraging technology to solve local and global issues. Authentic learning experiences that enable students to apply content knowledge, integrate concepts across disciplines, develop computational thinking skills, acquire and incorporate varied perspectives, and communicate with diverse audiences about the use and effects of computing prepares New Jersey students for college and careers."

    8.1.12.CS - Impacts of Computing

    - **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.CS.4:** Develop guidelines that convey systematic troubleshooting strategies that others can use to identify and fix errors.

    8.1.12.NI - Networks and the Internet

    - **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 - Impacts of Computing

    - **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 - Data & Analysis

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.

8.1.12.AP - Algorithms & Programming

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.

8.1.12.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.

Engineering Design

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.2: Create scaled engineering drawings for a new product or system and make modifications to increase optimization based on feedback.

8.2.12.ED.3: Evaluate several models of the same type of product and make recommendations for a new design based on a cost benefit analysis.

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.ED.6: Analyze the effects of changing resources when designing a specific product or system

Interaction of Technology and Humans

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.ITH.2: Propose an innovation to meet future demands supported by an analysis of the potential costs, benefits, trade-offs, and risks related to the use of the innovation.

8.2.12.ITH.3: Analyze the impact that globalization, social media, and access to open source technologies has had on innovation and on a society’s economy, politics, and culture.

Nature of Technology

8.2.12.NT.1: Explain how different groups can contribute to the overall design of a product.

8.2.12.NT.2: Redesign an existing product to improve form or function.

Effects of Technology on the Natural World

8.2.12.ETW.1: Evaluate ethical considerations regarding the sustainability of environmental resources that are used for the design, creation, and maintenance of a chosen product.

8.2.12.ETW.2: Synthesize and analyze data collected to monitor the effects of a technological product or system on the environment.

8.2.12.ETW.3: Identify a complex, global environmental or climate change issue, develop a systemic plan of investigation, and propose an innovative sustainable solution.

Ethics & Culture

8.2.12.EC.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.EC.2: Create scaled engineering drawings for a new product or system and make modification to increase optimization based on feedback

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.

8.2.12.EC.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.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.6: Analyze the effects of changing resources when designing a specific product or system.

2. **Career Ready Practices:** **https://www.state.nj.us/education/cccs/2014/career/CareerReadyPractices.pdf**
    - “Career Ready Practices describe the career-ready skills that all educators in all content areas should seek to develop in their students. They are practices that have been linked to increase college, career, and life success. Career Ready Practices should be taught and reinforced in all career exploration and preparation programs with increasingly higher levels of complexity and expectation as a student advances through a program of study.”

      CRP1 Act as a responsible and contributing citizen and employee.
      CRP2 Apply appropriate academic and technical skills.
      CRP3 Attend to personal health and financial well-being.
      CRP4 Communicate clearly and effectively and with reason.
      CRP5 Consider the environmental, social and economic impacts of decisions.
      CRP6 Demonstrate creativity and innovation.
      CRP7 Employ valid and reliable research strategies.
      CRP8 Utilize critical thinking to make sense of problems and persevere in solving them.
      CRP9 Model integrity, ethical leadership and effective management.
      CRP10 Plan education and career paths aligned to personal goals.
      CRP11 Use technology to enhance productivity.
      CRP12 Work productively in teams while using cultural global competence.
3. ***Standard 9.4 (Life Literacies and Key Skills)* of the *2020 NJSLS*:** **https://www.nj.gov/education/cccs/2020/2020%20NJSLS-CLKS.pdf**
    - “This standard outlines key literacies and technical skills such as critical thinking, global and cultural awareness, and technology literacy* that are critical for students to develop to live and work in an interconnected global economy.”

      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.CI.3:** Investigate new challenges and opportunities for personal growth, advancement, and transition
- **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
- **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.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.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.

4. ***LGBT and Disabilities Law: N.J.S.A. 18A:35-4.35*****:** **https://www.nj.gov/education/cccs/2020/2020%20NJSLS-CLKS.pdf**
   ○ A transformative approach to the inclusion of lessons and resources/texts on the contributions and issues concerning the LGBTQ+ population and people with disabilities will be implemented across all core subjects in accordance with state law: "A board of education shall include instruction on the political, economic, and social contributions of persons with disabilities and lesbian, gay, bisexual, and transgender people, in an appropriate place in the curriculum of middle school and high school students as part of the district's implementation of the New Jersey Student Learning Standards (N.J.S.A.18A:35-4.36). A board of education shall have policies and procedures in place pertaining to the selection of instructional materials to implement the requirements of N.J.S.A. 18A:35-4.35."
5. **Climate Change:** **https://www.nj.gov/education/cccs/2020/**
   ○ "Climate Change across all content areas, leveraging the passion students have shown for this critical issue and providing them opportunities to develop a deep understanding of the science behind the changes and to explore the solutions our world desperately needs"
6. **Acquisition/development/refinement of the higher-order critical thinking skills aligned with the *Revised Bloom's Taxonomy of Cognitive Objectives***

Section III: Curriculum Modifications

The *AP Computer Science Principles Curriculum* is subject to case-by-case modifications to support/advance the needs of all students, including special education students, English language learners, gifted students and those at risk of school failure. These modifications are based on Individualized Learning Programs (IEPs), recommendations made by the district's English Language Learners (ELL) coordinator, feedback from members of the Intervention & Referral Services Team (*I&RS*) for at-risk students, and 504 Plans.

Section IV: Preparation for Standardized Testing

Instruction in *AP Computer Science Principles* is aligned with the requirements of state and national standardized assessments, including the *NJSLA*, the *ACT*, the *PSAT* and the *SAT*. The *End of Marking Period Assessments* for *AP Computer Science Principles* also demonstrate alignment with the aforesaid standardized assessments.

Section V: Curriculum Pacing Guide

Curriculum Pacing Guide	
Course Title: AP Computer Science Principles	**Grade Level: 10-12**
Unit I: ***Digital Information***	September
Unit II: ***The Internet***	September - October
Unit III: ***Intro to App Design***	October
Unit IV: ***Variables, Conditionals, & Functions***	November
Unit V: ***Lists, Loops, & Traversals***	December
Unit VI: ***Algorithms***	January
Unit VII: ***Parameters, Return, & Libraries***	January
Unit VIII: ***Create PT Prep***	February
Unit IX: ***Data***	March
Unit X: ***Cybersecurity and Global Impact***	March - April
Unit XI: ***End of Course Project***	May - June

Section VI: Technology Skills

Not applicable to courses that are not core requirements.

Section VII: Texts and Instructional Resources

The following texts and instructional resources are employed in *AP Computer Science Principles*:

Student Resources:

- Code.org Studio: https://studio.code.org/home
- AP Students: https://apstudents.collegeboard.org/courses/ap-computer-science-principles
- My AP / AP Classroom: https://myap.collegeboard.org/login
- Advanced Placement, Computer Science Principles YouTube Channel: https://www.youtube.com/playlist?list=PLoGgviqq4844vbwcKegJgIxSQyVHDzSXT

Curricular Resources:

- AP Computer Science Principles Course and Exam Description: https://apcentral.collegeboard.org/pdf/ap-computer-science-principles-course-and-exam-description.pdf?SFMC_cid=EM318173-&rid=47696503
- AP Computer Science Principles Curricular Requirements https://drive.google.com/file/d/1oWRktMoZBqLzkD7jyyU5tzD--LdmoJzM/view?usp=sharing
- Code.org CS Principles Curriculum Guide 2020/2021: https://docs.google.com/document/d/1rDJhR4uCQN7y8V37xDejKUEKFeoUFAGVorPo8XMeR4I/edit
- *Common Sense Education* (www.commonsense.org)

Section VIII: Grading Formula and Assessment Modes

Marking period grades in *AP Computer Science Principles* are determined via a percentage weighting model that is comprised of the following grading categories:

Marking Periods 1, 2 and 4

Category	
Class Work	16 %
Homework	8 %
End of Unit Projects	28 %
Assessments	28 %
End of Marking Period Assessment	20 %

Marking Period 3

Category	
Class Work	20 %
Homework	10 %
End of Unit Projects	35 %
Assessments	35 %

Section IX: Unit Templates

The following *Unit Templates* have been established for the *AP Computer Science Principles Curriculum* by the *AP Computer Science Principles Instructional Team:*

Unit I: Digital Information

Students explore the way computers store and represent complex information like numbers, text, images, and sound. The unit begins with students investigating what it means to represent information, and challenges students to design their own representation systems. Students then learn the ideas behind real-world systems used to represent complex information. Later lessons focus on the challenges that arise from digitizing information, such as the need to compress it, or questions of intellectual property. The unit project emphasizes the profound impact digital information has on modern life.

AP Computational Thinking Practices

- Evaluate solution options.
- Implement and apply an algorithm.
- Explain how abstraction manages complexity
- Evaluate the use of computing based on legal and ethical factors.

Learning Objectives

- Explain how data can be represented using bits.
- Explain the consequences of using bits to represent data.
- Calculate the binary (base 2) equivalent of a positive integer (base 10) and vice versa.
- Compare and order binary numbers.
- Compare data compression algorithms to determine which is best in a particular context.
- Explain how the use of computing can raise legal and ethical concerns.

Standards/Core Ideas/Performance Expectations

- NJSLS
    - 8.1.12.CS.1
    - 8.1.12.CS.2
    - 8.1.12.CS.3
    - 8.1.12.CS.4
    - 8.1.12.IC.1
    - 8.1.12.IC.2
    - 8.1.12.IC.3
    - 8.1.12.DA.1
    - 8.1.12.DA.2
    - 8.1.12.DA.3
    - 8.1.12.DA.4
    - 8.1.12.DA.5
    - 8.1.12.DA.6
    - 8.2.12.ED.1
    - 8.2.12.ED.2
    - 8.2.12.ED.3
    - 8.2.12.ED.4
    - 8.2.12.ED.5
    - 8.2.12.ED.6
    - 8.2.12.ITH.1
    - 8.2.12.ITH.2
    - 8.2.12.ITH.3
    - 8.2.12.NT.1
    - 8.2.12.NT.2
- AP Computer Science Principles Curricular Requirements
    - CR 1, CR 2, CR 3, CR 9
- Integration of 21st Century Skills and Themes and Interdisciplinary Connections

- CRP1, CRP2, CRP3, CRP4, CRP5, CRP6, CRP7, CRP8, CRP9, CRP10, CRP11, CRP12
- 9.4.12.CI.1
- 9.4.12.CI.2
- 9.4.12.CI.3
- 9.4.12.CT.1
- 9.4.12.CT.2
- 9.4.12.CT.4
- 9.4.12.DC.1
- 9.4.12.DC.3
- 9.4.12.DC.4
- 9.4.12.DC.8
- 9.4.12.IML.3
- 9.4.12.IML.4
- 9.4.12.IML.7
- 9.4.12.TL.1
- 9.4.12.TL.2
- 9.4.12.TL.3
- 9.4.12.TL.4
- *LGBT and Disabilities Law*

Unit Essential Questions	Unit Enduring Understandings
● How can we use 1s and 0s to represent something complex like a video of the marching band playing a song? ● What app or computer software do you use most often and would have a hard time going without? How does this software solve a problem for you or benefit you? ● Are innovators responsible for the harmful effects of their computing innovations, even if those effects were unintentional? Why or why not?	● The way a computer represents data internally is different from the way the data are interpreted and displayed for the user. Programs are used to translate data into a representation more easily understood by people. ● The way a computer represents data internally is different from the way the data are interpreted and displayed for the user. Programs are used to translate data into a representation more easily understood by people. ● While computing innovations are typically designed to achieve a specific purpose, they may have unintended consequences.

Evidence of Learning		
Suggested Formative Assessment:	**Suggested Summative Assessment:**	**Resources Needed:**
● Classwork o Do-Nows o Journaling o Peer Feedback o Classroom Discussions o Pair Programming o Debugging o Unplugged and Plugged Activities o Exit Cards ● Homework ● AP Progress Checks ● Quizzes	● End of Unit Test ● End of Unit Project o Digital Information Dilemmas	● Laptop, Desktop, or Chromebook (Many assignments will not be accessible on a phone or tablet) ● Code.org Studio o Widgets o Internet Simulator o App Lab ● Worksheets ● Google Classroom ● Assortment of craft materials for constructing physical devices. Recommendations: cups, string/yarn, construction paper, flashlights, slinkies, noise makers, markers, and glue, etc.

Unit 2 - The Internet

Students learn how the Internet works and discuss its impacts on politics, culture, and the economy. This unit heavily features the Internet Simulator, a tool designed to let students see, use, and explore the way different layers of the internet work. Through a series of activities that build on one another, students investigate the problems the original designers of the internet had to solve and then "invent” their own solutions. At the conclusion of the unit, students research an "Internet Dilemma," both from the standpoint of its technical background and its impacts on different groups of people.

AP Computational Thinking Practices

- Evaluate solution options.
- Explain how computing systems work.
- Describe the impact of a computing innovation

Learning Objectives

- Explain how computing devices work together in a network.
- Explain how the Internet works
- Explain how data is sent through the Internet via packets.
- Describe the differences between the Internet and the World Wide Web.
- Describe the benefits of fault tolerance.
- Explain how a given system is fault-tolerant.
- Identify vulnerabilities to failure in a system.
- Describe issues that contribute to the digital divide.

Standards/Core Ideas/Performance Expectations

- NJSLS
    - 8.1.12.CS.1
    - 8.1.12.CS.2
    - 8.1.12.CS.3
    - 8.1.12.CS.4
    - 8.1.12.NI.1
    - 8.1.12.NI.2
    - 8.1.12.NI.3
    - 8.1.12.NI.4
    - 8.1.12.IC.1
    - 8.1.12.IC.2
    - 8.1.12.IC.3
    - 8.2.12.ED.1
    - 8.2.12.ED.2
    - 8.2.12.ED.3
    - 8.2.12.ED.4
    - 8.2.12.ED.5
    - 8.2.12.ED.6
    - 8.2.12.ITH.1
    - 8.2.12.ITH.2
    - 8.2.12.ITH.3
    - 8.2.12.NT.1
    - 8.2.12.NT.2
- AP Computer Science Principles Curricular Requirements
    - CR 1, CR 2, CR 3, CR 8, CR 10
- Integration of 21st Century Skills and Themes and Interdisciplinary Connections
    - CRP1, CRP2, CRP3, CRP4, CRP5, CRP6, CRP7, CRP8, CRP9, CRP10, CRP11, CRP12
    - 9.4.12.CI.1
    - 9.4.12.CI.2
    - 9.4.12.CI.3

○ 9.4.12.CT.1 ○ 9.4.12.CT.2 ○ 9.4.12.CT.4 ○ 9.4.12.DC.1 ○ 9.4.12.DC.3 ○ 9.4.12.DC.4 ○ 9.4.12.DC.8 ○ 9.4.12.IML.3 ○ 9.4.12.IML.4 ○ 9.4.12.IML.7 ○ 9.4.12.TL.1 ○ 9.4.12.TL.2 ○ 9.4.12.TL.3 ○ 9.4.12.TL.4 ○ *LGBT and Disabilities Law* ○ *Climate Change*

Unit Essential Questions	Unit Enduring Understandings
● Why are long text messages sometimes delivered out of order? ● When an Internet service outage occurs in a different part of your town or city, how are you still able to access the Internet? ● What app or computer software do you use most often and would have a hard time going without? How does this software solve a problem for you or benefit you? ● Are innovators responsible for the harmful effects of their computing innovations, even if those effects were unintentional? Why or why not?	● Computer systems and networks facilitate the transfer of data. ● While computing innovations are typically designed to achieve a specific purpose, they may have unintended consequences.

Evidence of Learning		
Suggested Formative Assessment:	**Suggested Summative Assessment:**	**Resources Needed:**
● Classwork o Do-Nows o Journaling o Peer Feedback o Classroom Discussions o Pair Programming o Debugging o Unplugged and Plugged Activities o Exit Cards ● Homework ● AP Progress Checks ● Quizzes	● End of Unit Test ● End of Unit Project o Internet Dilemmas	● Laptop, Desktop, or Chromebook (Many assignments will not be accessible on a phone or tablet) ● Code.org Studio o Widgets o Internet Simulator o App Lab ● Worksheets ● Google Classroom ● String for table groups to build a network connecting them or Students draw their network but don't actually build it

Unit 3: Intro to App Design

Students design their first app while learning both fundamental programming concepts and collaborative software development processes. Students work with partners to develop a simple app that teaches classmates about a topic of personal interest. Throughout the unit, they learn how to use Code.org's programming environment, App Lab, to design user interfaces and write simple event-driven programs. Along the way, students learn practices like debugging, pair programming, and collecting and responding to feedback, which they will be able to use throughout the course as they build ever more complex projects. The unit concludes with students sharing the apps they develop with their classmates.

AP Computational Thinking Practices

- Investigate the situation, context, or task.
- Determine and design an appropriate method or approach to achieve the purpose.
- Explain how collaboration affects the development of a solution.
- Generalize data sources through variables.
- Explain how a code segment or program functions.
- Collaborate in the development of solutions
- Acknowledge the intellectual property of others.

Learning Objectives

- Explain how computing innovations are improved through collaboration.
- Explain how computing innovations are developed by groups of people.
- Demonstrate effective interpersonal skills during collaboration.
- Describe the purpose of a computing innovation.
- Explain how a program or code segment functions.
- Identify input(s) to a program.
- Identify output(s) produced by a program.
- Develop a program using a development process.
- Design a program and its user interface.
- Describe the purpose of a code segment or program by writing documentation.
- Acknowledge code segments used from other sources.

Standards/Core Ideas/Performance Expectations

- NJSLS
  - 8.1.12.CS.1
  - 8.1.12.CS.2
  - 8.1.12.CS.3
  - 8.1.12.CS.4
  - 8.1.12.DA.1
  - 8.1.12.AP.1
  - 8.1.12.AP.2
  - 8.1.12.AP.3
  - 8.1.12.AP.4
  - 8.1.12.AP.5
  - 8.1.12.AP.6.
  - 8.1.12.AP.7
  - 8.1.12.AP.8
  - 8.1.12.AP.9
  - 8.2.12.ED.1
  - 8.2.12.ED.2
  - 8.2.12.ED.3
  - 8.2.12.ED.4
  - 8.2.12.ED.5
  - 8.2.12.ED.6

  - 8.2.12.ITH.1
  - 8.2.12.EC.1
  - 8.2.12.EC.2
  - 8.2.12.EC.3
  - 8.2.12.EC.4
  - 8.2.12.EC.5
  - 8.2.12.EC.6
- AP Computer Science Principles Curricular Requirements
  - CR 1, CR 2, CR 3, CR 4, CR 5, CR 6, CR 7, CR 8, CR 10
- Integration of 21st Century Skills and Themes and Interdisciplinary Connections
  - CRP1, CRP2, CRP3, CRP4, CRP5, CRP6, CRP7, CRP8, CRP9, CRP10, CRP11, CRP12
  - 9.4.12.CI.1
  - 9.4.12.CI.2
  - 9.4.12.CI.3
  - 9.4.12.CT.1
  - 9.4.12.CT.2
  - 9.4.12.CT.4
  - 9.4.12.DC.1
  - 9.4.12.DC.3
  - 9.4.12.DC.4
  - 9.4.12.DC.8
  - 9.4.12.IML.3
  - 9.4.12.IML.4
  - 9.4.12.IML.7
  - 9.4.12.TL.1
  - 9.4.12.TL.2
  - 9.4.12.TL.3
  - 9.4.12.TL.4

Unit Essential Questions	Unit Enduring Understandings
● How has working collaboratively with other students improved an overall project? ● What are some ways you can collect additional feedback on your program to use for improvements? ● What are some ways you currently plan your work before starting a project? ● What apps or programs have you stopped using because you didn't like the design of how you interacted with it?	● Incorporating multiple perspectives through collaboration improves computing innovations as they are developed. ● Developers create and innovate using an iterative design process that is user focused, that incorporates implementation/feedback cycles, and that leaves ample room for experimentation and risk-taking.

Evidence of Learning		
Suggested Formative Assessment:	**Suggested Summative Assessment:**	**Resources Needed:**
● Classwork o Do-Nows o Journaling o Peer Feedback o Classroom Discussions o Pair Programming o Debugging o Unplugged and Plugged Activities	● End of Unit Test ● End of Unit Project o Designing an App	● Laptop, Desktop, or Chromebook (Many assignments will not be accessible on a phone or tablet) ● Code.org Studio o Widgets o Internet Simulator o App Lab ● Worksheets ● Google Classroom

o Exit Cards ● Homework ● AP Progress Checks ● Quizzes		● A handful of LEGO® blocks for every pair students or Sticky notes, construction paper

Unit 4: Variables, Conditionals, and Functions

Students expand the types of apps they can create as they learn how to store information (variables), make decisions (conditionals), and better organize code (functions). Each programming topic is covered in a specific sequence of lessons that ask students to ‘Explore’ ideas through hands-on activities, ‘Investigate’ these ideas through guided code reading, ‘Practice’ with sample problems, and apply their understanding as they ‘Make’ a one-day scoped project. The entire unit concludes with a three-day open-ended project in which students must build an app that makes a recommendation about any topic they wish.

AP Computational Thinking Practices

- Determine and design an appropriate method or approach to achieve the purpose.
- Represent algorithmic processes without using a programming language.
- Implement and apply an algorithm.
- Generalize data sources through variables.
- Determine the result of code segments.
- Identify and correct errors in algorithms and programs, including error discovery through testing.

Learning Objectives

- For errors in an algorithm or program: Identify the error
- For errors in an algorithm or program: Correct the error
- Identify inputs and corresponding expected outputs or behaviors that can be used to check the correctness of an algorithm or program.
- Represent a value with a variable.
- Determine the value of a variable as a result of an assignment.
- Express an algorithm that uses sequencing without using a programming language.
- Represent a step-by-step algorithmic process using sequential code statements.
- Evaluate expressions that use arithmetic operators.
- For relationships between two variables, expressions, or values: Write expressions using relational operators
- For relationships between two variables, expressions, or values: Evaluate expressions that use relational operators
- For relationships between Boolean values: Write expressions using logic operators.
- For relationships between Boolean values: Evaluate expressions that use logic operators.
- Express an algorithm that uses selection without using a programming language.
- Write conditional statements.
- Determine the result of conditional statements.
- Write nested conditional statements.
- Determine the result of nested conditional statements.
- For generating random values: Write expressions to generate possible values.
- For generating random values: Evaluate expressions to determine the possible results.

Standards/Core Ideas/Performance Expectations

- NJSLS
    - 8.1.12.CS.1
    - 8.1.12.CS.2
    - 8.1.12.CS.3
    - 8.1.12.CS.4
    - 8.1.12.IC.1
    - 8.1.12.IC.2
    - 8.1.12.IC.3

  - 8.1.12.DA.1
  - 8.1.12.AP.1
  - 8.1.12.AP.2
  - 8.1.12.AP.3
  - 8.1.12.AP.4
  - 8.1.12.AP.5
  - 8.1.12.AP.6.
  - 8.1.12.AP.7
  - 8.1.12.AP.8
  - 8.1.12.AP.9
  - 8.2.12.ED.1
  - 8.2.12.ED.2
  - 8.2.12.ED.3
  - 8.2.12.ED.4
  - 8.2.12.ED.5
  - 8.2.12.ED.6
  - 8.2.12.ITH.1
  - 8.2.12.ITH.2
  - 8.2.12.ITH.3
  - 8.2.12.NT.1
  - 8.2.12.NT.2
  - 8.2.12.EC.1
  - 8.2.12.EC.2
  - 8.2.12.EC.3
  - 8.2.12.EC.4
  - 8.2.12.EC.5
  - 8.2.12.EC.6
- AP Computer Science Principles Curricular Requirements
  - CR 1, CR 2, CR 3, CR 4, CR 5, CR 6, CR 7, CR 8
- Integration of 21st Century Skills and Themes and Interdisciplinary Connections
  - CRP1, CRP2, CRP3, CRP4, CRP5, CRP6, CRP7, CRP8, CRP9, CRP10, CRP11, CRP12
  - 9.4.12.CI.1
  - 9.4.12.CI.2
  - 9.4.12.CI.3
  - 9.4.12.CT.1
  - 9.4.12.CT.2
  - 9.4.12.CT.4
  - 9.4.12.DC.1
  - 9.4.12.DC.3
  - 9.4.12.DC.4
  - 9.4.12.DC.8
  - 9.4.12.IML.3
  - 9.4.12.IML.4
  - 9.4.12.IML.7
  - 9.4.12.TL.1
  - 9.4.12.TL.2
  - 9.4.12.TL.3
  - 9.4.12.TL.4
  - *LGBT and Disabilities Law*
  - *Climate Change*

Unit Essential Questions	**Unit Enduring Understandings**

● What are some ways you currently plan your work before starting a project? ● What apps or programs have you stopped using because you didn’t like the design of how you interacted with it? ● How can we store data in a program to solve problems? ● What might happen if you completed the steps in your regular morning routine to get ready and go to school in a different order? How might the reordering affect the decisions you make each morning? ● How do video games group the different actions for a player based on what key is pressed on the keyboard or controller? How do apps group different actions together based on user interaction, such as pressing buttons?	● Developers create and innovate using an iterative design process that is user focused, that incorporates implementation/feedback cycles, and that leaves ample room for experimentation and risk-taking. ● To find specific solutions to generalizable problems, programmers represent and organize data in multiple ways. ● The way statements are sequenced and combined in a program determines the computed result. Programs incorporate iteration and selection constructs to represent repetition and make decisions to handle varied input values. ● Programmers break down problems into smaller and more manageable pieces. By creating procedures and leveraging parameters, programmers generalize processes that can be reused. Procedures allow programmers to draw upon existing code that has already been tested, allowing them to write programs more quickly and with more confidence.

Evidence of Learning		
Suggested Formative Assessment:	**Suggested Summative Assessment:**	**Resources Needed:**
● Classwork o Do-Nows o Journaling o Peer Feedback o Classroom Discussions o Pair Programming o Debugging o Unplugged and Plugged Activities o Exit Cards ● Homework ● AP Progress Checks ● Quizzes	● End of Unit Test ● End of Unit Project o Decision Maker App	● Laptop, Desktop, or Chromebook (Many assignments will not be accessible on a phone or tablet) ● Code.org Studio o Widgets o Internet Simulator o App Lab ● Worksheets ● Google Classroom ● Plastic bags, sticky notes, dry erase markers or Envelopes

Unit 5: Lists, Loops, and Traversals

Students learn to build apps that use and process lists of information. Like the previous unit, students learn the core concepts of lists, loops, and traversals through a series of EIPM lesson sequences. Later in the unit, students are introduced to tools that allow them to import tables of real-world data to help further power the types of apps they can make. At the conclusion of the unit, students complete a week-long project in which they must design an app around a goal of their choosing that uses one of these data sets.

AP Computational Thinking Practices

- Investigate the situation, context, or task.
- Evaluate solution options.
- Represent algorithmic processes without using a programming language.
- Implement and apply an algorithm.
- Generalize data sources through variables.
- Use abstraction to manage complexity in a program.
- Explain how abstraction manages complexity

- Determine the result of code segments.

Learning Objectives

- Represent a list or string using a variable.
- Develop data abstraction using lists to store multiple elements.
- Explain how the use of data abstraction manages complexity in program code
- Evaluate expressions that manipulate strings.
- Express an algorithm that uses iteration without using a programming language.
- Write iteration statements.
- Determine the result or side effect of iteration statements.
- For list operations: Write expressions that use list indexing and list procedures.
- Evaluate expressions that use list indexing and list procedures
- For algorithms involving elements of a list: Write iteration statements to traverse a list
- Determine the result of an algorithm that includes list traversals
- For simulations: Explain how computers can be used to represent real-world phenomena or outcomes.
- Compare simulations with real-world contexts.

Standards/Core Ideas/Performance Expectations

- NJSLS
  - 8.1.12.CS.1
  - 8.1.12.CS.2
  - 8.1.12.CS.3
  - 8.1.12.CS.4
  - 8.1.12.IC.1
  - 8.1.12.IC.2
  - 8.1.12.IC.3
  - 8.1.12.DA.1
  - 8.1.12.DA.2
  - 8.1.12.DA.3
  - 8.1.12.DA.4
  - 8.1.12.DA.5
  - 8.1.12.DA.6
  - 8.1.12.AP.1
  - 8.1.12.AP.2
  - 8.1.12.AP.3
  - 8.1.12.AP.4
  - 8.1.12.AP.5
  - 8.1.12.AP.6.
  - 8.1.12.AP.7
  - 8.1.12.AP.8
  - 8.1.12.AP.9
  - 8.2.12.ED.1
  - 8.2.12.ED.2
  - 8.2.12.ED.3
  - 8.2.12.ED.4
  - 8.2.12.ED.5
  - 8.2.12.ED.6
  - 8.2.12.ITH.1
  - 8.2.12.ITH.2
  - 8.2.12.ITH.3
  - 8.2.12.NT.1
  - 8.2.12.NT.2
  - 8.2.12.ETW.1.
  - 8.2.12.ETW.2
  - 8.2.12.ETW.3

  - 8.2.12.EC.1
  - 8.2.12.EC.2
  - 8.2.12.EC.3
  - 8.2.12.EC.4
  - 8.2.12.EC.5
  - 8.2.12.EC.6
- AP Computer Science Principles Curricular Requirements
  - CR 1, CR 2, CR 3, CR 4, CR 5, CR 6, CR 7, CR 8, CR 9
- Integration of 21st Century Skills and Themes and Interdisciplinary Connections
  - CRP1, CRP2, CRP3, CRP4, CRP5, CRP6, CRP7, CRP8, CRP9, CRP10, CRP11, CRP12
  - 9.4.12.CI.1
  - 9.4.12.CI.2
  - 9.4.12.CI.3
  - 9.4.12.CT.1
  - 9.4.12.CT.2
  - 9.4.12.CT.4
  - 9.4.12.DC.1
  - 9.4.12.DC.3
  - 9.4.12.DC.4
  - 9.4.12.DC.8
  - 9.4.12.IML.3
  - 9.4.12.IML.4
  - 9.4.12.IML.7
  - 9.4.12.TL.1
  - 9.4.12.TL.2
  - 9.4.12.TL.3
  - 9.4.12.TL.4
  - *LGBT and Disabilities Law*
  - *Climate Change*

Unit Essential Questions	Unit Enduring Understandings
● How can we store data in a program to solve problems? ● What might happen if you completed the steps in your regular morning routine to get ready and go to school in a different order? How might the reordering affect the decisions you make each morning? ● How do video games group the different actions for a player based on what key is pressed on the keyboard or controller? How do apps group different actions together based on user interaction, such as pressing buttons?	● To find specific solutions to generalizable problems, programmers represent and organize data in multiple ways. ● The way statements are sequenced and combined in a program determines the computed result. Programs incorporate iteration and selection constructs to represent repetition and make decisions to handle varied input values. ● Programmers break down problems into smaller and more manageable pieces. By creating procedures and leveraging parameters, programmers generalize processes that can be reused. Procedures allow programmers to draw upon existing code that has already been tested, allowing them to write programs more quickly and with more confidence.

Evidence of Learning

Suggested Formative Assessment:	**Suggested Summative Assessment:**	**Resources Needed:**
● Classwork o Do-Nows o Journaling o Peer Feedback o Classroom Discussions o Pair Programming	● End of Unit Test ● End of Unit Project o Hackathon	● Laptop, Desktop, or Chromebook (Many assignments will not be accessible on a phone or tablet) ● Code.org Studio o Widgets o Internet Simulator o App Lab

o Debugging o Unplugged and Plugged Activities o Exit Cards ● Homework ● AP Progress Checks ● Quizzes		● Worksheets ● Google Classroom ● Plastic bags, gallon-sized plastic bags, sticky notes, dry erase markers, tape or Envelopes

Unit 6: Algorithms

Students learn to design and analyze algorithms to understand how they work and why some algorithms are considered more efficient than others. This short unit is entirely unplugged, and features hands-on activities that help students get an intuitive sense of how quickly different algorithms run and the pros and cons of different algorithms. Later in the unit, students explore concepts like undecidable problems and parallel and distributed computing.

AP Computational Thinking Practices

- Investigate the situation, context, or task.
- Evaluate solution options.
- Represent algorithmic processes without using a programming language.
- Implement and apply an algorithm.

Learning Objectives

- Compare multiple algorithms to determine if they yield the same side effect or result.
- Create algorithms
- Combine and modify existing algorithms
- For binary search algorithms: Determine the number of iterations required to find a value in a data set.
- For binary search algorithms: Explain the requirements necessary to complete a binary search.
- For determining the efficiency of an algorithm: Explain the difference between algorithms that run in reasonable time and those that do not.
- For determining the efficiency of an algorithm: Identify situations where a heuristic solution may be more appropriate.
- Explain the existence of undecidable problems in computer science.
- For sequential, parallel, and distributed computing: Compare problem solutions.
- For sequential, parallel, and distributed computing: Determine the efficiency of solutions.
- Describe benefits and challenges of parallel and distributed computing.

Standards/Core Ideas/Performance Expectations

- NJSLS
  - 8.1.12.IC.1
  - 8.1.12.IC.2
  - 8.1.12.IC.3
  - 8.1.12.AP.1
  - 8.1.12.AP.2
  - 8.1.12.AP.5
  - 8.1.12.AP.7
  - 8.1.12.AP.8
  - 8.1.12.AP.9
  - 8.2.12.ED.5
  - 8.2.12.ED.6
- AP Computer Science Principles Curricular Requirements
  - CR 1, CR 2, CR 3, CR 4, CR 5, CR 8, CR 10
- Integration of 21st Century Skills and Themes and Interdisciplinary Connections
  - CRP1, CRP2, CRP3, CRP4, CRP5, CRP6, CRP7, CRP8, CRP9, CRP10, CRP11, CRP12

- ○ 9.4.12.CI.1
- ○ 9.4.12.CI.2
- ○ 9.4.12.CI.3
- ○ 9.4.12.CT.1
- ○ 9.4.12.CT.2
- ○ 9.4.12.CT.4
- ○ 9.4.12.DC.1
- ○ 9.4.12.DC.3
- ○ 9.4.12.DC.4
- ○ 9.4.12.DC.8
- ○ 9.4.12.IML.3
- ○ 9.4.12.IML.4
- ○ 9.4.12.IML.7
- ○ 9.4.12.TL.1
- ○ 9.4.12.TL.2
- ○ 9.4.12.TL.3
- ○ 9.4.12.TL.4

Unit Essential Questions	Unit Enduring Understandings
● What might happen if you completed the steps in your regular morning routine to get ready and go to school in a different order? How might the reordering affect the decisions you make each morning? ● What types of problems can be solved more easily with a computer, and what types can be solved more easily without a computer? Why? ● What are the benefits of dividing tasks among group members? ● Is there a point where adding another group member would not make completing the task faster? Why?	● The way statements are sequenced and combined in a program determines the computed result. Programs incorporate iteration and selection constructs to represent repetition and make decisions to handle varied input values. ● The way statements are sequenced and combined in a program determines the computed result. Programs incorporate iteration and selection constructs to represent repetition and make decisions to handle varied input values. ● There exist problems that computers cannot solve, and even when a computer can solve a problem, it may not be able to do so in a reasonable amount of time. ● Parallel and distributed computing leverage multiple computers to more quickly solve complex problems or process large data sets.

Evidence of Learning

Suggested Formative Assessment:	Suggested Summative Assessment:	Resources Needed:
● Classwork o Do-Nows o Journaling o Peer Feedback o Classroom Discussions o Pair Programming o Debugging o Unplugged and Plugged Activities o Exit Cards ● Homework ● AP Progress Checks ● Quizzes	● End of Unit Test	● Laptop, Desktop, or Chromebook (Many assignments will not be accessible on a phone or tablet) ● Code.org Studio o Widgets o Internet Simulator o App Lab ● Worksheets ● Google Classroom ● Sticky notes or Scraps of paper ● Decks of cards or Any item that could be combined into two categories (e.g. change with even / odd year)

Unit 7: Parameters, Return, and Libraries

Students learn how to design clean and reusable code that can be shared with a single classmate or the entire world. In the beginning of the unit, students are introduced to the concepts of parameters and return, which allow for students to design functions that implement an algorithm. In the second half of the unit, students learn how to design libraries of functions that can be packaged up and shared with others. The unit concludes with students designing their own small library of functions that can be used by a classmate.

AP Computational Thinking Practices

- Implement and apply an algorithm.
- Use abstraction to manage complexity in a program.
- Explain how abstraction manages complexity.
- Determine the result of code segments.

Learning Objectives

- Write statements to call procedures.
- Determine the result or effect of a procedure call.
- Explain how the use of procedural abstraction manages complexity in a program.
- Develop procedural abstractions to manage complexity in a program by writing procedures.
- Select appropriate libraries or existing code segments to use in creating new programs.

Standards/Core Ideas/Performance Expectations

- NJSLS
    - 8.1.12.CS.1
    - 8.1.12.CS.2
    - 8.1.12.CS.3
    - 8.1.12.CS.4
    - 8.1.12.IC.1
    - 8.1.12.IC.2
    - 8.1.12.IC.3
    - 8.1.12.AP.1
    - 8.1.12.AP.2
    - 8.1.12.AP.3
    - 8.1.12.AP.4
    - 8.1.12.AP.5
    - 8.1.12.AP.6.
    - 8.1.12.AP.7
    - 8.1.12.AP.8
    - 8.1.12.AP.9
    - 8.2.12.ED.1
    - 8.2.12.ED.2
    - 8.2.12.ED.3
    - 8.2.12.ED.4
    - 8.2.12.ED.5
    - 8.2.12.ED.6
    - 8.2.12.ITH.1
    - 8.2.12.ITH.2
    - 8.2.12.ITH.3
    - 8.2.12.NT.1
    - 8.2.12.NT.2
    - 8.2.12.EC.1
    - 8.2.12.EC.2
    - 8.2.12.EC.3
    - 8.2.12.EC.4
    - 8.2.12.EC.5
    - 8.2.12.EC.6
- AP Computer Science Principles Curricular Requirements

  - CR 1, CR 2, CR 3, CR 4, CR 5, CR 6, CR 7, CR 8, CR 10
- Integration of 21st Century Skills and Themes and Interdisciplinary Connections
  - CRP1, CRP2, CRP3, CRP4, CRP5, CRP6, CRP7, CRP8, CRP9, CRP10, CRP11, CRP12
  - 9.4.12.CI.1
  - 9.4.12.CI.2
  - 9.4.12.CI.3
  - 9.4.12.CT.1
  - 9.4.12.CT.2
  - 9.4.12.CT.4
  - 9.4.12.DC.1
  - 9.4.12.DC.3
  - 9.4.12.DC.4
  - 9.4.12.DC.8
  - 9.4.12.IML.3
  - 9.4.12.IML.4
  - 9.4.12.IML.7
  - 9.4.12.TL.1
  - 9.4.12.TL.2
  - 9.4.12.TL.3
  - 9.4.12.TL.4

Unit Essential Questions	Unit Enduring Understandings
● How do video games group the different actions for a player based on what key is pressed on the keyboard or controller? How do apps group different actions together based on user interaction, such as pressing buttons?	● Programmers break down problems into smaller and more manageable pieces. By creating procedures and leveraging parameters, programmers generalize processes that can be reused. Procedures allow programmers to draw upon existing code that has already been tested, allowing them to write programs more quickly and with more confidence.

Evidence of Learning		
Suggested Formative Assessment:	**Suggested Summative Assessment:**	**Resources Needed:**
● Classwork o Do-Nows o Journaling o Peer Feedback o Classroom Discussions o Pair Programming o Debugging o Unplugged and Plugged Activities o Exit Cards ● Homework ● AP Progress Checks ● Quizzes	● End of Unit Test ● End of Unit Project ● Make a Library	● Laptop, Desktop, or Chromebook (Many assignments will not be accessible on a phone or tablet) ● Code.org Studio o Widgets o Internet Simulator o App Lab ● Worksheets ● Google Classroom ● Sticky notes, envelopes, plastic bags, file folders or Scraps of paper, folders made of a folded sheet of paper, etc.

Unit 8: Create PT Prep
In this unit, students practice and complete the Create Performance Task (PT), starting with a series of activities that ensure they understand the College Board requirements of the Create PT, which they have practiced throughout the year. Subsequently, students are given at least 12 class hours in which to complete the Create PT.
AP Computational Thinking Practices

- All to date.

Learning Objectives

- Design and evaluate computational solutions for a purpose.
- Develop and implement algorithms.
- Develop programs that incorporate abstractions.
- Evaluate and test algorithms and programs.
- Contribute to an inclusive, safe, collaborative, and ethical computing culture.

Standards/Core Ideas/Performance Expectations

- NJSLS
    - 8.1.12.CS.1
    - 8.1.12.CS.2
    - 8.1.12.CS.3
    - 8.1.12.CS.4
    - 8.1.12.NI.1
    - 8.1.12.NI.2
    - 8.1.12.NI.3
    - 8.1.12.NI.4
    - 8.1.12.IC.1
    - 8.1.12.IC.2
    - 8.1.12.IC.3
    - 8.1.12.DA.1
    - 8.1.12.DA.2
    - 8.1.12.DA.3
    - 8.1.12.DA.4
    - 8.1.12.DA.5
    - 8.1.12.DA.6
    - 8.1.12.AP.1
    - 8.1.12.AP.2
    - 8.1.12.AP.3
    - 8.1.12.AP.4
    - 8.1.12.AP.5
    - 8.1.12.AP.6.
    - 8.1.12.AP.7
    - 8.1.12.AP.8
    - 8.1.12.AP.9
    - 8.2.12.ED.1
    - 8.2.12.ED.2
    - 8.2.12.ED.3
    - 8.2.12.ED.4
    - 8.2.12.ED.5
    - 8.2.12.ED.6
    - 8.2.12.ITH.1
    - 8.2.12.ITH.2
    - 8.2.12.ITH.3
    - 8.2.12.NT.1
    - 8.2.12.NT.2
    - 8.2.12.ETW.1.
    - 8.2.12.ETW.2
    - 8.2.12.ETW.3
    - 8.2.12.EC.1
    - 8.2.12.EC.2
    - 8.2.12.EC.3
    - 8.2.12.EC.4

- 8.2.12.EC.5
- 8.2.12.EC.6

- AP Computer Science Principles Curricular Requirements
  - CR 1, CR 2, CR 3, CR 4, CR 5, CR 6, CR 7, CR 11
- Integration of 21st Century Skills and Themes and Interdisciplinary Connections
  - CRP1, CRP2, CRP3, CRP4, CRP5, CRP6, CRP7, CRP8, CRP9, CRP10, CRP11, CRP12
  - 9.4.12.CI.1
  - 9.4.12.CI.2
  - 9.4.12.CI.3
  - 9.4.12.CT.1
  - 9.4.12.CT.2
  - 9.4.12.CT.4
  - 9.4.12.DC.1
  - 9.4.12.DC.3
  - 9.4.12.DC.4
  - 9.4.12.DC.8
  - 9.4.12.IML.3
  - 9.4.12.IML.4
  - 9.4.12.IML.7
  - 9.4.12.TL.1
  - 9.4.12.TL.2
  - 9.4.12.TL.3
  - 9.4.12.TL.4

Unit Essential Questions	Unit Enduring Understandings
● How can I apply the principles and skills of computer science to my life to solve problems?	● Computer science is interdisciplinary ● Computer science is collaborative ● Computer science is innovative ● Computer science applications can benefit society

Evidence of Learning		
Suggested Formative Assessment:	**Suggested Summative Assessment:**	**Resources Needed:**
● Classwork o Do-Nows o Journaling o Peer Feedback o Classroom Discussions o Pair Programming o Debugging o Unplugged and Plugged Activities o Exit Cards ● Homework	● Create Performance Task for AP Test	● Laptop, Desktop, or Chromebook (Many assignments will not be accessible on a phone or tablet) ● Code.org Studio o Widgets o Internet Simulator o App Lab ● Worksheets ● Google Classroom

Unit 9: Data
Students explore and visualize datasets from a wide variety of topics as they hunt for patterns and try to learn more about the world around them from the data. Once again, students work with datasets in App Lab, but are now asked to make use of a data visualizer tool that assists students in finding data patterns. They learn how different types of visualizations can be used to better understand the patterns contained in datasets and how to use visualizations when investigating hypotheses. At the conclusion of the unit, students learn about the impacts of data analysis on the world

around them and complete a final project in which they must uncover and present a data investigation they've completed independently.

AP Computational Thinking Practices

- Explain how collaboration affects the development of a solution.
- Implement and apply an algorithm.
- Explain how knowledge can be generated from data.
- Describe the impact of gathering data.
- Evaluate the use of computing based on legal and ethical factors.

Learning Objectives

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

Standards/Core Ideas/Performance Expectations

- NJSLS
  - 8.1.12.CS.1
  - 8.1.12.CS.2
  - 8.1.12.CS.3
  - 8.1.12.CS.4
  - 8.1.12.IC.1
  - 8.1.12.IC.2
  - 8.1.12.IC.3
  - 8.1.12.DA.1
  - 8.1.12.DA.2
  - 8.1.12.DA.3
  - 8.1.12.DA.4
  - 8.1.12.DA.5
  - 8.1.12.DA.6
  - 8.2.12.ED.1
  - 8.2.12.ED.2
  - 8.2.12.ED.3
  - 8.2.12.ED.4
  - 8.2.12.ED.5
  - 8.2.12.ED.6
  - 8.2.12.ITH.1
  - 8.2.12.ITH.2
  - 8.2.12.ITH.3
  - 8.2.12.NT.1
  - 8.2.12.NT.2
  - 8.2.12.ETW.1.
  - 8.2.12.ETW.2
  - 8.2.12.ETW.3
  - 8.2.12.EC.1
  - 8.2.12.EC.2
  - 8.2.12.EC.3
  - 8.2.12.EC.4
  - 8.2.12.EC.5
  - 8.2.12.EC.6
- AP Computer Science Principles Curricular Requirements

- CR 1, CR 2, CR 3, CR 4, CR 5, CR 7, CR 8, CR 9, CR 10
- Integration of 21st Century Skills and Themes and Interdisciplinary Connections
  - CRP1, CRP2, CRP3, CRP4, CRP5, CRP6, CRP7, CRP8, CRP9, CRP10, CRP11, CRP12
  - 9.4.12.CI.1
  - 9.4.12.CI.2
  - 9.4.12.CI.3
  - 9.4.12.CT.1
  - 9.4.12.CT.2
  - 9.4.12.CT.4
  - 9.4.12.DC.1
  - 9.4.12.DC.3
  - 9.4.12.DC.4
  - 9.4.12.DC.8
  - 9.4.12.IML.3
  - 9.4.12.IML.4
  - 9.4.12.IML.7
  - 9.4.12.TL.1
  - 9.4.12.TL.2
  - 9.4.12.TL.3
  - 9.4.12.TL.4
  - *LGBT and Disabilities Law*
  - *Climate Change*

Unit Essential Questions	Unit Enduring Understandings
● How can we use 1s and 0s to represent something complex like a video of the marching band playing a song? ● How can you predict the attendance at a school event using data gathered from social media? ● When is it more appropriate to use a computer to analyze data than to complete the analysis by hand? ● What app or computer software do you use most often and would have a hard time going without? How does this software solve a problem for you or benefit you? ● Are innovators responsible for the harmful effects of their computing innovations, even if those effects were unintentional? Why or why not?	● Programs can be used to process data, which allows users to discover information and create new knowledge. ● While computing innovations are typically designed to achieve a specific purpose, they may have unintended consequences.

Evidence of Learning		
Suggested Formative Assessment:	**Suggested Summative Assessment:**	**Resources Needed:**
● Classwork o Do-Nows o Journaling o Peer Feedback o Classroom Discussions o Pair Programming o Debugging	● End of Unit Test ● End of Unit Project o Tell a Data Story	● Laptop, Desktop, or Chromebook (Many assignments will not be accessible on a phone or tablet) ● Code.org Studio o Widgets o Internet Simulator o App Lab ● Worksheets

o Unplugged and Plugged Activities o Exit Cards ● Homework ● AP Progress Checks ● Quizzes		● Google Classroom

Unit 10: Cybersecurity and Global Impacts

Students research and debate current events at the intersection of data, public policy, law, ethics, and societal impact in the final unit of the course. This unit is built around a simulated "future school" conference in which students must take on the persona of a stakeholder in a school setting and propose and debate technological innovations that could improve schools. Throughout the unit, students learn about the privacy and security risks of many computing innovations, and learn about the ways some of these risks can be mitigated. Students complete their Explore Curricular Requirement as part of this project as they investigate at least three computing innovations, then discuss and debate many others with their classmates. At the conclusion of the unit, the class holds a conference in which teams present their overall vision for a school of the future and the computing innovations that would power it.

AP Computational Thinking Practices

- Describe the impact of a computing innovation.
- Describe the impact of gathering data.
- Evaluate the use of computing based on legal and ethical factors.

Learning Objectives

- Explain how an effect of a computing innovation can be both beneficial and harmful.
- Explain how a computing innovation can have an impact beyond its intended purpose.
- Describe the risks to privacy from collecting and storing personal data on a computer system.
- Explain how computing resources can be protected and can be misused.
- Explain how unauthorized access to computing resources is gained.

Standards/Core Ideas/Performance Expectations

- NJSLS
  - 8.1.12.CS.1
  - 8.1.12.CS.2
  - 8.1.12.CS.3
  - 8.1.12.CS.4
  - 8.1.12.NI.1
  - 8.1.12.NI.2
  - 8.1.12.NI.3
  - 8.1.12.NI.4
  - 8.1.12.IC.1
  - 8.1.12.IC.2
  - 8.1.12.IC.3
  - 8.2.12.ED.1
  - 8.2.12.ED.2
  - 8.2.12.ED.3
  - 8.2.12.ED.4
  - 8.2.12.ED.5
  - 8.2.12.ED.6
  - 8.2.12.ITH.1
  - 8.2.12.ITH.2
  - 8.2.12.ITH.3
  - 8.2.12.NT.1

-
    - 8.2.12.NT.2
    - 8.2.12.ETW.1.
    - 8.2.12.ETW.2
    - 8.2.12.ETW.3
    - 8.2.12.EC.1
    - 8.2.12.EC.2
    - 8.2.12.EC.3
    - 8.2.12.EC.4
    - 8.2.12.EC.5
    - 8.2.12.EC.6
- AP Computer Science Principles Curricular Requirements
    - CR 1, CR 2, CR 3, CR 4, CR 5, CR 6, CR 7, CR 8, CR 9, CR 10
- Integration of 21st Century Skills and Themes and Interdisciplinary Connections
    - CRP1, CRP2, CRP3, CRP4, CRP5, CRP6, CRP7, CRP8, CRP9, CRP10, CRP11, CRP12
    - 9.4.12.CI.1
    - 9.4.12.CI.2
    - 9.4.12.CI.3
    - 9.4.12.CT.1
    - 9.4.12.CT.2
    - 9.4.12.CT.4
    - 9.4.12.DC.1
    - 9.4.12.DC.3
    - 9.4.12.DC.4
    - 9.4.12.DC.8
    - 9.4.12.IML.3
    - 9.4.12.IML.4
    - 9.4.12.IML.7
    - 9.4.12.TL.1
    - 9.4.12.TL.2
    - 9.4.12.TL.3
    - 9.4.12.TL.4
    - *LGBT and Disabilities Law*
    - *Climate Change*

Unit Essential Questions	Unit Enduring Understandings
● What app or computer software do you use most often and would have a hard time going without? How does this software solve a problem for you or benefit you? ● Are innovators responsible for the harmful effects of their computing innovations, even if those effects were unintentional? Why or why not? ● What data are generated by smartphones, and what are they being used for?	● While computing innovations are typically designed to achieve a specific purpose, they may have unintended consequences. ● The use of computing innovations may involve risks to personal safety and identity.

Evidence of Learning		
Suggested Formative Assessment:	**Suggested Summative Assessment:**	**Resources Needed:**
● Classwork o Do-Nows o Journaling o Peer Feedback	● End of Unit Tests ● End of Unit Project o Innovation Simulation	● Laptop, Desktop, or Chromebook (Many assignments will not be accessible on a phone or tablet) ● Code.org Studio

o Classroom Discussions o Pair Programming o Debugging o Unplugged and Plugged Activities o Exit Cards ● Homework ● AP Progress Checks ● Quizzes		o Widgets o Internet Simulator o App Lab ● Worksheets ● Google Classroom

Unit 11: End of Course Project

Collaboration allows computer scientists to improve their products. During this unit, students will be encouraged to collaborate with many different students, especially those who have different perspectives than their own. Collaborating with people who have a different perspective helps to reveal blind spots they might have due to their own backgrounds and experiences.

Learning Objectives

- Design and evaluate computational solutions for a purpose.
- Develop and implement algorithms.
- Develop programs that incorporate abstractions.
- Evaluate and test algorithms and programs.
- Investigate computing innovations.
- Contribute to an inclusive, safe, collaborative, and ethical computing culture.

Standards/Core Ideas/Performance Expectations

- NJSLS
    - 8.1.12.CS.1
    - 8.1.12.CS.2
    - 8.1.12.CS.3
    - 8.1.12.CS.4
    - 8.1.12.NI.1
    - 8.1.12.NI.2
    - 8.1.12.NI.3
    - 8.1.12.NI.4
    - 8.1.12.IC.1
    - 8.1.12.IC.2
    - 8.1.12.IC.3
    - 8.1.12.DA.1
    - 8.1.12.DA.2
    - 8.1.12.DA.3
    - 8.1.12.DA.4
    - 8.1.12.DA.5
    - 8.1.12.DA.6
    - 8.1.12.AP.1
    - 8.1.12.AP.2
    - 8.1.12.AP.3
    - 8.1.12.AP.4
    - 8.1.12.AP.5
    - 8.1.12.AP.6.
    - 8.1.12.AP.7
    - 8.1.12.AP.8

- 8.1.12.AP.9
- 8.2.12.ED.1
- 8.2.12.ED.2
- 8.2.12.ED.3
- 8.2.12.ED.4
- 8.2.12.ED.5
- 8.2.12.ED.6
- 8.2.12.ITH.1
- 8.2.12.ITH.2
- 8.2.12.ITH.3
- 8.2.12.NT.1
- 8.2.12.NT.2
- 8.2.12.ETW.1.
- 8.2.12.ETW.2
- 8.2.12.ETW.3
- 8.2.12.EC.1
- 8.2.12.EC.2
- 8.2.12.EC.3
- 8.2.12.EC.4
- 8.2.12.EC.5
- 8.2.12.EC.6

- AP Computer Science Principles Curricular Requirements
  - CR 1, CR 2, CR 3, CR 4, CR 5, CR 6, CR 7, CR 8, CR 9, CR 10
- Integration of 21st Century Skills and Themes and Interdisciplinary Connections
  - CRP1, CRP2, CRP3, CRP4, CRP5, CRP6, CRP7, CRP8, CRP9, CRP10, CRP11, CRP12
  - 9.4.12.CI.1
  - 9.4.12.CI.2
  - 9.4.12.CI.3
  - 9.4.12.CT.19.4.12.CT.2
  - 9.4.12.CT.4
  - 9.4.12.DC.1
  - 9.4.12.DC.3
  - 9.4.12.DC.4
  - 9.4.12.DC.8
  - 9.4.12.IML.3
  - 9.4.12.IML.4
  - 9.4.12.IML.7
  - 9.4.12.TL.1
  - 9.4.12.TL.2
  - 9.4.12.TL.3
  - 9.4.12.TL.4
  - *LGBT and Disabilities Law*
  - *Climate Change*

Unit Essential Questions	Unit Enduring Understandings
● How can I apply the principles and skills of computer science to my life to solve problems?	● Computer science is interdisciplinary ● Computer science is collaborative ● Computer science is innovative ● Computer science applications can benefit society

Evidence of Learning		
Suggested Formative Assessment:	**Suggested Summative Assessment:**	**Resources Needed:**
● Classwork o Peer Feedback	● End of Unit Project	● Laptop, Desktop, or Chromebook (Many assignments will not be accessible on a phone or tablet)

o Classroom Discussions o Pair Programming o Debugging		● Code.org Studio o Widgets o Internet Simulator o App Lab

Section X: Unit Reflection

The *AP Computer Science Principles Instructional Team* must confer upon the completion of each instructional unit in *AP Computer Science Principles* and rate the degrees to which the instructional units meet performance criteria established by the New Jersey Department of Education using the *Unit Reflection Form*. Completed *Unit Reflection Form*s must be submitted to the Department Supervisor for approval upon completion of curriculum implementation with a complementing list of suggested modifications to the *AP Computer Science Principles Curriculum*.

Lesson Activities:	Strongly	Moderately	Weakly
Foster student use of technology as a tool to develop critical thinking, creativity and innovation skills;			
Are challenging and require higher order thinking and problem solving skills;			
Allow for student choice;			
Provide scaffolding for acquiring targeted knowledge/skills;			
Integrate global perspectives;			
Integrate 21st century skills;			
Provide opportunities for interdisciplinary connection and transfer of knowledge and skills;			
Are varied to address different student learning styles and preferences;			
Are differentiated based on student needs;			
Are student-centered with teacher acting as a facilitator and co-learner during the teaching and learning process;			
Provide means for students to demonstrate knowledge and skills and progress in meeting learning goals and objectives;			
Provide opportunities for student reflection and self-assessment;			
Provide data to inform and adjust instruction to better meet the varying needs of learners;			