INTRODUCTION TO COMPUTER SCIENCE [HONORS]
GRADES 9-12

EWING PUBLIC SCHOOLS
2099 Pennington Road
Ewing, NJ 08618

Board Approval Date: July 25, 2016
Produced by: EHS Staff
Michael Nitti
Superintendent

In accordance with The Ewing Public Schools' Policy 2230, Course Guides, this curriculum has been reviewed and found to be in compliance with all policies and all affirmative action criteria.

Table of Contents

Course Description and Rationale

Introduction to Computer Science is designed to introduce students to the central ideas of computer science, to instill ideas and practices of computational thinking and to have students engage in activities that show how computing changes the world. The course is rigorous and rich in computational content, includes computational and critical thinking skills and engages students in the creative aspects of the field. Through both its content and pedagogy, this course aims to appeal to a broad audience. This intellectually rich and engaging course emphasizes three key themes that help students build a solid understanding and facility with computing and computational thinking:

Theme 1: Creativity as a method of computational design
Theme 2: Use of technology as a means for problem solving and innovation
Theme 3: Relevance of technology and computing to people and society

These understandings are important, if not integral, to being part of a well-educated and informed citizenry.

The first theme of the Computer Science Principles course is its focus on creativity. The computational thinking practices and big ideas that follow hint at the creative nature of computing, yet alone they cannot truly convey the importance of creativity in this course. It's not enough for students to know that "computing requires creativity." Rather, students must actually be creative: creating artifacts that they want to show off to their friends and family, using simulation to explore questions that interest them and designing and implementing solutions employing the iterative and sometimes messy process that artists, writers, computer scientists and engineers use to translate ideas into tangible form.

A second theme is the course's use of technology as a means for solving computational problems and exploring creative endeavors, rather than a focus on a specific tool or language. To that end, the course highlights programming as one of the seven big ideas of computer science, because programming is among the creative processes that help transform ideas into reality. Programming is a tool students use to explore concepts and create exciting and personally relevant artifacts. In contrast to traditional college introductory computer science courses the Computer Science Principles course does not focus on and is not organized around a specific language. The instructor of the course selects one or more languages, based on appropriateness for a specific project or problem and according to guidelines provided as part of the course specification. Language specifics are taught only to the extent that students need them to produce their programs. Similarly, data and the use of computational tools to analyze and study data, is another of the big ideas of computer science, as data plays an incredibly important role in so many aspects of our lives. Students in this course work with "big-data" - they analyze it, visualize it, draw conclusions from trends in it - but the course itself does not specify particular tools for these explorations.

A third theme that helps the course appeal to a broad audience is the course's focus on people and society, not just on machines and systems. Students in a Computer Science Principles course explore computer science's relevance to and impact on the world today. They investigate the innovations in other fields that computing has made possible. They examine the ethical implications of new computing technologies. They perform activities that develop their communication and collaboration skills. Students in this course work individually and collaboratively to solve problems. They talk and write about their solutions, the importance of

these problems and their impact on the world. This curriculum framework specifies the course curriculum: the content, practices, thinking and skills central to the discipline of computer science. Through this novel content with implications for engaging pedagogy, students will experience the joy and beauty that permeates computing: They will not only experience the sense of community from connecting with friends on social networks, but they will understand many aspects of the software and algorithms that make these social networks possible. They will not only use algorithms, but also create them and experience the “ah ha!” moment when an algorithm finally makes sense. They will not simply run programs; they will experience the thrill of constructing a program and seeing it work, as well as the pride of creating something for oneself, one’s family or friends or for the world.

The course follows a block semester schedule, with students meeting daily for 88 minutes. The course content is arranged into seven units of study:

Unit 1: Creativity and Computing: Preview and Setup
Unit 2: Mobile Computers and Mobile Apps
Unit 3: Creating Graphics and Images Bit by Bit
Unit 4: Animations, Simulations and Modeling
Unit 5: Algorithms and Procedural Abstractions
Unit 6: Using and Analyzing Data and Information
Unit 7: Communication Through The Internet

This course addresses the New Jersey Student Learning Standards in Technology. The specific technology standards addressed will be listed in each unit description. The design of this course also incorporates reading and writing about technical content and the design of computer programs and thus addresses standards in both Common Core State Standards (CCSS) in English and Language Arts for Literacy in Technical Subjects and the Engineering Standards within the Next Generation Science Standards (NGSS).

Unit 1: Creativity and Computing: Preview and Setup [1 Week]

Why Is This Unit Important?

Unit 1 of the course provides a brief overview of the Mobile CSP curriculum, emphasizing its main theme: learning the principles of computer science while building socially useful mobile apps. The hands-on work focuses on setting up the student's environment, including their programming environment and online portfolios. Students are led through the process of creating or opening their Gmail account, registering on the App Inventor site and setting up their Google sites portfolio. Their portfolios will be used to display and share all of their written work for the course. Students are provided a brief introduction to blocks-based programming by having them work through a series of increasingly challenging Blocky Maze problems. And they are given a brief introduction to the Blown to Bits book, which is used as a reading resource throughout the course.

Enduring Understandings:

- Computing enables people to use creative development processes when using computing tools and techniques to create computational artifacts for creative expression of ideas or to solve problems.
- Algorithms are precise sequences of instructions for processes which can be executed by a computer and are implemented using programming language.
- People write programs to execute algorithms.
- Cybersecurity is an important concern for the internet and the systems built on it.
- Computing enhances communication, interaction and cognition.
- Computing has global effects – both beneficial and harmful – on people and society.
- Computer innovations influence and are influenced by the economic, social and cultural contexts in which they are designed and used.

Essential Questions:

- What is the Mobile CS Principles course?
- What is graphical blocks-based programming?
- Why is it important to study the impact of computing technology?

Acquired Knowledge:

- Identify Google sites portfolio as an example of cloud computing.
- Create a Google sites portfolio that they will use during the course to post their work.

Acquired Skills:

- Clearly communicate with a global audience about personal ideas.
- Understand how humans and computers interact through different languages.

Major Assessments:

- Portfolio
- Reading and Homework assignments

- Labs
- Projects
- Oral Presentations
- Quizzes and Exams
- Self-Check and Live Coding Exercises

Suggested Learning Experiences and Instructional Activities:

In-Class Activities and Laboratory Experiences:

- Computer Logins, Network Logins, File Structures, Mazes, Blocky, Algorithms, Control Structures, Google Accounts, Google Apps Introduction, Google Sites

Instructional Materials/Technology Connections:

- https://ram8647.appspot.com/mobileCSP/course (This is the main link to mobileCSP which provides links to all needed resources.)

List of Applicable Standards Covered in This Unit:

Content Standards/NJSLS Technology Standards:

- 8.1.12.A.1: Create a personal digital portfolio which reflects personal and academic interests, achievements and career aspirations by using a variety of digital tools and resources.
- 8.1.12.A.3: Collaborate in online courses, learning communities, social networks or virtual worlds to discuss a resolution to a problem or issue.
- 8.1.12.E.3: Use a programming language to solve problems or accomplish a task (e.g., robotic functions, website designs, applications and games).
- 8.1.12.E.3: Use appropriate terms in conversation (e.g., troubleshooting, peripherals, diagnostic software, GUI, abstraction, variables, data types and conditional statements).
- 8.2.12.A.1: Propose an innovation to meet future demands supported by an analysis of the potential full costs, benefits, trade-offs and risks, related to the use of the innovation.
- 8.2.12.A.2: Analyze a current technology and the resources used, to identify the trade-offs in terms of availability, cost, desirability and waste.

Unit 2: Mobile Computers and Mobile Apps [(2 Weeks]

Why Is This Unit Important?

Unit 2 provides an introduction to the App Inventor programming platform and the course's first programming project, the I Have a Dream app, a sound board app. Students are introduced to App Inventor's event-driven programming model. Students first work through a guided tutorial that plays an excerpt of a Martin Luther King speech and are then presented with several exercises that challenge them to extend their understanding by solving problems on their own, working in pairs. This is followed later in the unit by several creative mini projects where students are invited to express their own ideas by developing their own computational artifacts. Students are also introduced to several important CS Principles themes and topics. Two lessons focus on hardware and software concepts. The big idea of abstraction is introduced. Students get their first look at binary numbers learning how to count in binary and how to view number systems such as binary, hexadecimal and decimal, as instances of the higher order abstraction of a positional number system.

Enduring Understandings:

- Computing enables people to use creative development processes when using computing tools and techniques to create computational artifacts for creative expression of ideas or to solve problems.
- A variety of abstractions built upon binary sequences can be used to represent all digital data.
- Multiple levels of abstraction are used to write programs or to create other computational artifacts.
- Models and simulations use abstraction to generate new understanding and knowledge.
- People use computer programs to process information to gain insight and knowledge.
- There are trade-offs when representing information as digital data.
- Programs can be developed to solve problems (to help people, organizations or society); for creative expression; to satisfy personal curiosity or to create new knowledge.
- People write programs to execute algorithms.
- Programming is facilitated by appropriate abstractions.
- Programs are developed, maintained and used by people for different purposes.
- Programming uses mathematical and logical concepts.
- Computing enhances communication, interaction and cognition.

Essential Questions:

- How does one use App Inventor and event-driven programming to build a mobile app?
- What are the various hardware and software abstractions that make up a modern digital computer?
- What is the binary number system that underlies all digital representation?

Acquired Knowledge:

- Create a computational artifact for creative expression.
- Create a computational artifact using computing tools and techniques to solve a problem.

- Create a new computational artifact by combining or modifying existing artifacts.
- Describe the variety of abstractions used to represent data.
- Explain how binary sequences are used to represent digital data.
- Identify multiple levels of abstractions being used when writing programs.
- Develop a program for creative expression, to satisfy personal curiosity or to create new knowledge.
- Explain how programs implement algorithms.
- Evaluate the correctness of a program.
- Explain how computing innovations affect communication, interaction and cognition.

Acquired Skills:

- Use the creative development processes to solve problems.
- Use binary sequences to represent digital data.
- Use simulations to foster new understandings and knowledge.
- Apply mathematical and logical concepts

Major Assessments:

- Portfolio
- Reading and Homework assignments
- Labs
- Projects
- Oral Presentations
- Quizzes and Exams
- Self-Check and Live Coding Exercises

Suggested Learning Experiences and Instructional Activities:

In-Class Activities and Laboratory Experiences:

- I Have a Dream Tutorial Parts 1 & 2, Mobile Apps and Mobile Devices, I Have a Dream Projects (Soundboard), Blown to Bits Chapter 1, Where Is North: A Compass App, Paint Pot Tutorial Part 1 & 2 and Projects

Instructional Materials/Technology Connections:

- https://ram8647.appspot.com/mobileCSP/course (This is the main link to mobileCSP which provides links to all needed resources.)

List of Applicable Standards Covered in This Unit:

Content Standards/NJSLS Technology Standards:

- 8.1.12.A.1: Create a personal digital portfolio which reflects personal and academic interests, achievements and career aspirations by using a variety of digital tools and resources.
- 8.1.12.A.3: Collaborate in online courses, learning communities, social networks or virtual worlds to discuss a resolution to a problem or issue.

- 8.1.12.E.3: Use a programming language to solve problems or accomplish a task (e.g., robotic functions, website designs, applications and games).
- 8.1.12.E.3: Use appropriate terms in conversation (e.g., troubleshooting, peripherals, diagnostic software, GUI, abstraction, variables, data types and conditional statements).
- 8.2.12.A.1: Propose an innovation to meet future demands supported by an analysis of the potential full costs, benefits, trade-offs and risks, related to the use of the innovation.
- 8.2.12.A.2: Analyze a current technology and the resources used, to identify the trade-offs in terms of availability, cost, desirability and waste.

Unit 3: Creating Graphics and Images Bit by Bit [2 Weeks]

Why Is This Unit Important?

Unit 3 extends the student's mobile programming toolkit to several new App Inventor components and introduces a number of new programming concepts, including the concept of a variables, lists and data abstraction. The main app in this unit, The Paint Pot app, a computational version of finger painting, focuses on App Inventor's drawing and painting features and related topics from the CS Principles framework. The app is presented in four parts each of which is followed by a set of creative project exercises and challenges. This unit also introduces two other apps: Magic 8 Ball app, which provides a first introduction to lists and Map Tour, which demonstrates how to incorporate external data into a mobile app. Unit 3 also extends the student's understanding of binary number system and introduces students to the idea of a bit as the fundamental unit of data. Through a number of hands-on and interactive activities students explore how bits are used to represent images and how redundant parity bits can be used to detect simple data transmission errors. These lessons are complemented nicely by a Blown to Bits reading that focuses on digital documents, including how information can be hidden inside images and other digital documents.

Enduring Understandings:

- Computing enables people to use creative development processes when using computing tools and techniques to create computational artifacts for creative expression of ideas or to solve a problem.
- A variety of abstractions built upon binary sequences can be used to represent all digital data.
- Multiple levels of abstraction are used to write programs or to create other computational artifacts.
- Programs can be developed to solve problems (to help people, organizations or society); for creative expression; to satisfy personal curiosity or to create new knowledge.
- People write programs to execute algorithms.
- Programs are developed, maintained and used by people for different purposes.
- Programming uses mathematical and logical concepts.
- Computing enhances communication, interaction and cognition.

Essential Questions:

- How can binary numbers be used to represent all digital data?
- How can algorithms be used to compress data?
- How do variables of both simple and structured data, such as, lists, enable us manage the complexity of a programming?

Acquired Knowledge:

- Create a computational artifact for creative expression.
- Create a computational artifact using computing tools and techniques to solve a problem.
- Create a new computational artifact by combining or modifying existing artifacts.
- Identify multiple levels of abstractions being used when writing programs.

- Develop a program for creative expression, to satisfy personal curiosity or to create new knowledge.
- Employ appropriate mathematical and logical concepts in programming.
- Explain how computing innovations affect communication, interaction and cognition.

Acquired Skills:

- Write programs using multiple levels of abstraction.
- Use algorithms to compress data.
- How computing innovations affect communication, interaction and cognition.

Major Assessments:

- Project: Programming Performance Task #1 (Creativity, Abstraction, Algorithms and Programming)
- Portfolio
- Reading and Homework assignments
- Labs
- Projects
- Oral Presentations
- Quizzes and Exams
- Self-Check and Live Coding Exercises

Major Assessments:

- Portfolio
- Reading and Homework assignments
- Labs
- Projects
- Oral Presentations
- Quizzes and Exams
- Self-Check and Live Coding Exercises

Suggested Learning Experiences and Instructional Activities:

In-Class Activities and Laboratory Experiences:

- Blown to Bits Chapter 3, Magic 8 Ball Tutorial, Map Tour Tutorial

Instructional Materials/Technology Connections:

- https://ram8647.appspot.com/mobileCSP/course (This is the main link to mobileCSP which provides links to all needed resources.)

List of Applicable Standards Covered in This Unit:

Content Standards/NJSLS Technology Standards:

- 8.1.12.A.1: Create a personal digital portfolio which reflects personal and academic interests, achievements and career aspirations by using a variety of digital tools and resources.
- 8.1.12.A.3: Collaborate in online courses, learning communities, social networks or virtual worlds to discuss a resolution to a problem or issue.
- 8.1.12.E.3: Use a programming language to solve problems or accomplish a task (e.g., robotic functions, website designs, applications and games).
- 8.1.12.E.3: Use appropriate terms in conversation (e.g., troubleshooting, peripherals, diagnostic software, GUI, abstraction, variables, data types and conditional statements).
- 8.2.12.A.1: Propose an innovation to meet future demands supported by an analysis of the potential full costs, benefits, trade-offs and risks, related to the use of the innovation.
- 8.2.12.A.2: Analyze a current technology and the resources used, to identify the trade-offs in terms of availability, cost, desirability and waste.

Unit 4: Animations, Simulations and Modeling [3 Weeks]

Why Is This Unit Important?

Unit 4 focuses on animation, simulation and modeling. The Android Mash app introduces the idea of computer simulation with a computational version of the traditional Whack-a-Mole game. The Coin Flip app, which extends over several lessons, introduces the concept of modeling. The activities in Unit 4 build toward EU 2.3 as students learn that models use abstractions, such as a pseudo random number generator (PRNG), to represent real word situations, in this case, the flipping of a coin; EU 3.3 as students learn how PRNG algorithms are used to model randomness inside a computer, such as with the Coin Flip app; EU 7.1 as students extend the app model to represent different types of coins, including a biased coin and a three--sided coin. This is followed by an experimental lesson where an app that repeatedly "flips" a coin is used to assess the quality of App Inventor's PRNG; EU 7.3 as students learn how one's privacy is impacted by developing technology and computing innovations; and EU 7.4 as students learn the economic, social and cultural effects of computing innovations, such as real world models of the weather and the solar system.

Enduring Understandings:

- Computing enables people to use creative development processes when using computing tools and techniques to create computational artifacts for creative expression of ideas or to solve a problem.
- Computing can extend traditional forms of human expression and experience.
- Multiple levels of abstraction are used to write programs or to create other computational artifacts.
- Use models and simulations to represent phenomena.
- There are trade-offs when representing information as digital data.
- Algorithms are precise sequences of instructions for processes that can be executed by a computer and are implemented using programming languages.
- Programs can be developed to solve problems (to help people, organizations or society); for creative expression; to satisfy personal curiosity or to create new knowledge.
- Programming is facilitated by appropriate abstractions.
- Programming uses mathematical and logical concepts.
- Computing enhances communication, interaction and cognition.
- Computing has global effects - both beneficial and harmful - on people and society
- Computing innovations influence and are influenced by the economic, social and cultural contexts in which they are designed and used.

Essential Questions:

- How do computers use simulation and modeling to represent real world phenomena?
- Why is randomness important and how is it modeled inside a computer?
- In what ways does simulation and modeling extend our knowledge and benefit society?

Acquired Knowledge:

- Create a computational artifact for creative expression.
- Create a new computational artifact by combining or modifying existing artifacts.

- Use computing tools and techniques for creative expression.
- Develop an abstraction when writing a program or creating other computational artifacts.
- Use models and simulations to represent phenomena.
- Use models and simulations to formulate, refine and test hypotheses.
- Analyze how data representation, storage, security and transmission of data involve computational manipulation of information.
- Develop an algorithm for implementation in a program.
- Express an algorithm in a language.
- Develop a correct program to solve problems.
- Use abstraction to manage complexity in programs.
- Employ appropriate mathematical and logical concepts in programming.
- Explain how computing innovations affect communication, interaction and cognition.
- Analyze the beneficial and harmful effects of computing.
- Explain the connections between computing and economic, social and cultural contexts.

Acquired Skills:

- Use computers to represent real world problems.
- Use the pseudo random number generator to represent real world issues.
- Understand how privacy is impacted.
- Use technology and understand the effects of computing innovations on real world models.

Major Assessments:

- Portfolio
- Reading and Homework assignments
- Labs
- Projects
- Oral Presentations
- Quizzes and Exams
- Self-Check and Live Coding Exercises

Suggested Learning Experiences and Instructional Activities:

In-Class Activities and Laboratory Experiences:

- Android Mash, Android Mash Projects, Coin Flip Simulation, Coin Flip Experiment, Pseudo Random Numbers, Coin Flip Simulation Projects, Real World Models, Blown to Bits Chapter 2: Privacy

Instructional Materials/Technology Connections:

- https://ram8647.appspot.com/mobileCSP/course (This is the main link to mobileCSP which provides links to all needed resources.)

List of Applicable Standards Covered in This Unit:

Content Standards/NJSLS Technology Standards:

- 8.1.12.A.1: Create a personal digital portfolio which reflects personal and academic interests, achievements and career aspirations by using a variety of digital tools and resources.
- 8.1.12.A.3: Collaborate in online courses, learning communities, social networks or virtual worlds to discuss a resolution to a problem or issue.
- 8.1.12.E.3: Use a programming language to solve problems or accomplish a task (e.g., robotic functions, website designs, applications and games).
- 8.1.12.E.3: Use appropriate terms in conversation (e.g., troubleshooting, peripherals, diagnostic software, GUI, abstraction, variables, data types and conditional statements).
- 8.2.12.A.1: Propose an innovation to meet future demands supported by an analysis of the potential full costs, benefits, trade-offs and risks, related to the use of the innovation.
- 8.2.12.A.2: Analyze a current technology and the resources used, to identify the trade-offs in terms of availability, cost, desirability and waste.

Unit 5: Algorithms and Procedural Abstractions [2 Weeks]

Why Is This Unit Important?

In Unit 5, algorithms and procedures are examined in more detail. The Logo apps introduce the concept of procedural abstraction and students learn to define and use procedures -- named blocks of code that perform a specific task. By encapsulating the algorithms into named procedures and introducing parameters to help generalize the algorithms, students are led to see the advantages of procedural abstraction. In addition to designing and testing their own algorithms, students are also provided an introduction into the analysis of algorithms. Algorithm efficiency is examined for searching and sorting algorithms, which are analyzed both experimentally and through mathematical concepts such as functions and graphs. The impact section of this unit focuses on the impact that Web searching algorithms have had on our lives. The activities completed in Unit 5 build toward EU 2.2, EU 4.1, EU 4.2, EU 5.3 and EU 5.5 by focusing on abstraction, algorithms and programming concepts.

Enduring Understandings:

- Multiple levels of abstraction are used to write programs or to create other computational artifacts.
- Algorithms are precise sequences of instructions for processes that can be executed by a computer and are implemented using programming languages.
- Algorithms can solve many but not all problems.
- Programs can be developed to solve problems (to help people, organizations or society); for creative expression; to satisfy personal curiosity or to create new knowledge.
- Programming is facilitated by appropriate abstractions.
- Programs are developed, maintained and used by people for different purposes.
- Computing enhances communication, interaction and cognition.

Essential Questions:

- What is the Internet, how is it built and how does it function?
- What aspects of the Internet's design and development have helped it scale and flourish?
- How is cyber security impacting the ever increasing number of Internet users?

Acquired Knowledge:

- Develop an abstraction when writing a program or creating other computational artifacts.
- Use multiple levels of abstraction to write programs.
- Develop an algorithm for implementation in a program.
- Express an algorithm in a language.
- Explain the difference between algorithms that run in a reasonable time and those that do not run in a reasonable time.
- Explain the difference between solvable and unsolvable problems in computer science.
- Explain the existence of undecidable problems in computer science.
- Evaluate algorithms analytically and empirically for efficiency, correctness and clarity.
- Develop a correct program to solve problems.
- Use abstraction to manage complexity in programs.

- Evaluate the correctness of a program.
- Explain how computing innovations affect communication, interaction and cognition.
- Explain how people participate in a problem solving process that scales.

Acquired Skills:

- Know how the internet functions.
- Use named blocks of code to perform a specific task.
- Analyze the efficiency of algorithms.

Major Assessments:

- Portfolio
- Reading and Homework assignments
- Labs
- Projects
- Oral Presentations
- Quizzes and Exams
- Self-Check and Live Coding Exercises

Suggested Learning Experiences and Instructional Activities:

In-Class Activities and Laboratory Experiences:

- What Is an Algorithm?, Logo Part 1, Logo Part 2, Search Algorithms, Sorting Algorithms, Analyzing Algorithms, The Pong Game, Limits of Algorithms, Debugging Pong, Blown to Bits Chapter 4: Web Searches

Instructional Materials/Technology Connections:

- https://ram8647.appspot.com/mobileCSP/course (This is the main link to mobileCSP which provides links to all needed resources.)

List of Applicable Standards Covered in This Unit:

Content Standards/NJSLS Technology Standards:

- 8.1.12.A.1: Create a personal digital portfolio which reflects personal and academic interests, achievements and career aspirations by using a variety of digital tools and resources.
- 8.1.12.A.3: Collaborate in online courses, learning communities, social networks or virtual worlds to discuss a resolution to a problem or issue.
- 8.1.12.E.3: Use a programming language to solve problems or accomplish a task (e.g., robotic functions, website designs, applications and games).
- 8.1.12.E.3: Use appropriate terms in conversation (e.g., troubleshooting, peripherals, diagnostic software, GUI, abstraction, variables, data types and conditional statements).
- 8.2.12.A.1: Propose an innovation to meet future demands supported by an analysis of the potential full costs, benefits, trade-offs and risks, related to the use of the innovation.
- 8.2.12.A.2: Analyze a current technology and the resources used, to identify the trade-offs in terms of availability, cost, desirability and waste.

Unit 6: Using and Analyzing Data and Information [3 Weeks]

Why Is This Unit Important?

Unit 6 focuses on various aspects of using and manipulating Data, both within mobile apps and on the Web and Internet. The App Inventor lessons in this unit focus on different types of programming data, including variables and structured data, such as lists and databases. Students build apps that involve persistent data, data that persists from one instance of the app to another and learn how to share data online by using simple Application Programming Interfaces (APIs), such as the Google Fusion table API. This unit's CS Principles lessons build toward EU 3.1, EU 3.2, EU 7.1, EU 7.2 and EU 7.5 by focusing on the concept of Big Data and its growing importance and its impact on society. Students are also introduced to some of the algorithms for processing massive datasets.

Enduring Understandings:

- Computing enables people to use creative development processes when using computing tools and techniques to create computational artifacts for creative expression of ideas or to solve a problem.
- People use computer programs to process information to gain insight and knowledge.
- Computing facilitates exploration and the discovery of connections in information.
- There are trade-offs when representing information as digital data.
- Programs can be developed to solve problems (to help people, organizations or society); for creative expression; to satisfy personal curiosity or to create new knowledge.
- People write programs to execute algorithms.
- Programming is facilitated by appropriate abstractions.
- Programming uses mathematical and logical concepts.
- Computing enables innovation in nearly every field.
- Computing has global effects - both beneficial and harmful - on people and society.

Essential Questions:

- How does continuous access to large amounts of data change how people and organizations make decisions?
- How do computers put things in order and find things in a list?
- What is the connection between data, information, knowledge and wisdom?

Acquired Knowledge:

- Create a computational artifact for creative expression.
- Create a computational artifact using computing tools and techniques to solve a problem.
- Create a new computational artifact by combining or modifying existing artifacts.
- Use computers to process information, find patterns and test hypotheses about digitally processed information to gain insight and knowledge.
- Explain the insight and knowledge gained from digitally processed data by using appropriate visualizations, notation and precise language.
- Use large data sets to explore and discover information and knowledge.

- Analyze how data representation, storage, security and transmission of data involve computational manipulation of information.
- Develop a correct program to solve problems.
- Explain how programs implement algorithms.
- Use abstraction to manage complexity in programs.
- Employ appropriate mathematical and logical concepts in programming.
- Explain how computing has impacted innovations in other fields.
- Analyze the beneficial and harmful effects of computing.

Acquired Skills:

- What order is information stored.
- How to find information in a list.
- The impact and importance of Big Data on society.

Major Assessments:

- Portfolio
- Reading and Homework assignments
- Labs
- Projects
- Oral Presentations
- Quizzes and Exams
- Self-Check and Live Coding Exercises

Suggested Learning Experiences and Instructional Activities:

In-Class Activities and Laboratory Experiences:

- Presidents Quiz, Presidents Quiz Projects, Blown to Bits Chapter 6: Who Owns the Bits?, Lists of Lists, Persistent Data, Sharing Data on the Web, Data Persistence Projects, Big Data

Instructional Materials/Technology Connections:

- https://ram8647.appspot.com/mobileCSP/course (This is the main link to mobileCSP which provides links to all needed resources.)

List of Applicable Standards Covered in This Unit:

Content Standards/NJSLS Technology Standards:

- 8.1.12.A.1: Create a personal digital portfolio which reflects personal and academic interests, achievements and career aspirations by using a variety of digital tools and resources.
- 8.1.12.A.3: Collaborate in online courses, learning communities, social networks or virtual worlds to discuss a resolution to a problem or issue.
- 8.1.12.E.3: Use a programming language to solve problems or accomplish a task (e.g., robotic functions, website designs, applications and games).

- 8.1.12.E.3: Use appropriate terms in conversation (e.g., troubleshooting, peripherals, diagnostic software, GUI, abstraction, variables, data types and conditional statements).
- 8.2.12.A.1: Propose an innovation to meet future demands supported by an analysis of the potential full costs, benefits, trade-offs and risks, related to the use of the innovation.
- 8.2.12.A.2: Analyze a current technology and the resources used, to identify the trade-offs in terms of availability, cost, desirability and waste.

Unit 7: Communication Through the Internet [2 Weeks]

Why Is This Unit Important?

Unit 7 focuses on the Internet, one of the big ideas in computer science. The App Inventor lessons in this unit show different ways to use the internet in apps, including the ability to send text messages over Wifi, finding directions via the Google Maps API. The CS Principles lessons focus on the Internet, how it works, how it enables innovation and collaboration and security concerns for using it.

Enduring Understandings:

- Computing enables people to use creative development processes when using computing tools and techniques to create computational artifacts for creative expression of ideas or to solve a problem.
- Programming is facilitated by appropriate abstractions.
- The Internet is a network of autonomous systems.
- Characteristics of the Internet influence the systems built on it.
- Cybersecurity is an important concern for the Internet and the systems built on it.
- Computing enhances communication, interaction and cognition.
- Computing has a global affect – both beneficial and harmful – on people and society.
- Computing innovations influence and are influenced by the economic, social and cultural contexts in which they are designed and used.

Essential Questions:

- Is the internet the same as the World Wide Web?
- How does an email go from one computer to the other?
- What activities and tools support a secure Web experience?

Acquired Knowledge:

- Create a computational artifact using computing tools and techniques to solve a problem.
- Use abstraction to manage complexity in programs.
- Explain the abstractions in the Internet and how the Internet functions.
- Explain characteristics of the Internet and the systems built on it.
- Explain how the characteristics of the Internet influence the systems built on it.
- Identify existing cybersecurity concerns and potential options that address these issues with the Internet and the systems built on it.
- Explain how computing innovations affect communication, interaction and cognition.
- Analyze the beneficial and harmful effects of computing.
- Explain the connections between computing and economic, social and cultural contexts.

Acquired Skills:

- How to utilize the internet with Apps.
- Send text messages using Wifi.

- Get directions using Google Maps.
- The different tools and activities to promote a secure Web experience.

Major Assessments:

- Portfolio
- Reading and Homework assignments
- Labs
- Projects
- Oral Presentations
- Quizzes and Exams
- Self-Check and Live Coding Exercises

Suggested Learning Experiences and Instructional Activities:

In-Class Activities and Laboratory Experiences:

- What Is the Internet?, No Texting While Busy Tutorial, Cloud Computing and Ethics, How the Internet Works, My Directions Tutorial, Cryptography Basics, Cryptography: Securing the Internet, Blown to Bits Chapter 5: Cryptography, Social Aware App: Broadcast Hub Tutorial

Instructional Materials/Technology Connections:

- https://ram8647.appspot.com/mobileCSP/course (This is the main link to mobileCSP which provides links to all needed resources.)

List of Applicable Standards Covered in This Unit:

Content Standards/NJSLS Technology Standards:

- 8.1.12.A.1: Create a personal digital portfolio which reflects personal and academic interests, achievements and career aspirations by using a variety of digital tools and resources.
- 8.1.12.A.3: Collaborate in online courses, learning communities, social networks or virtual worlds to discuss a resolution to a problem or issue.
- 8.1.12.E.3: Use a programming language to solve problems or accomplish a task (e.g., robotic functions, website designs, applications and games).
- 8.1.12.E.3: Use appropriate terms in conversation (e.g., troubleshooting, peripherals, diagnostic software, GUI, abstraction, variables, data types and conditional statements).
- 8.2.12.A.1: Propose an innovation to meet future demands supported by an analysis of the potential full costs, benefits, trade-offs and risks, related to the use of the innovation.
- 8.2.12.A.2: Analyze a current technology and the resources used, to identify the trade-offs in terms of availability, cost, desirability and waste.