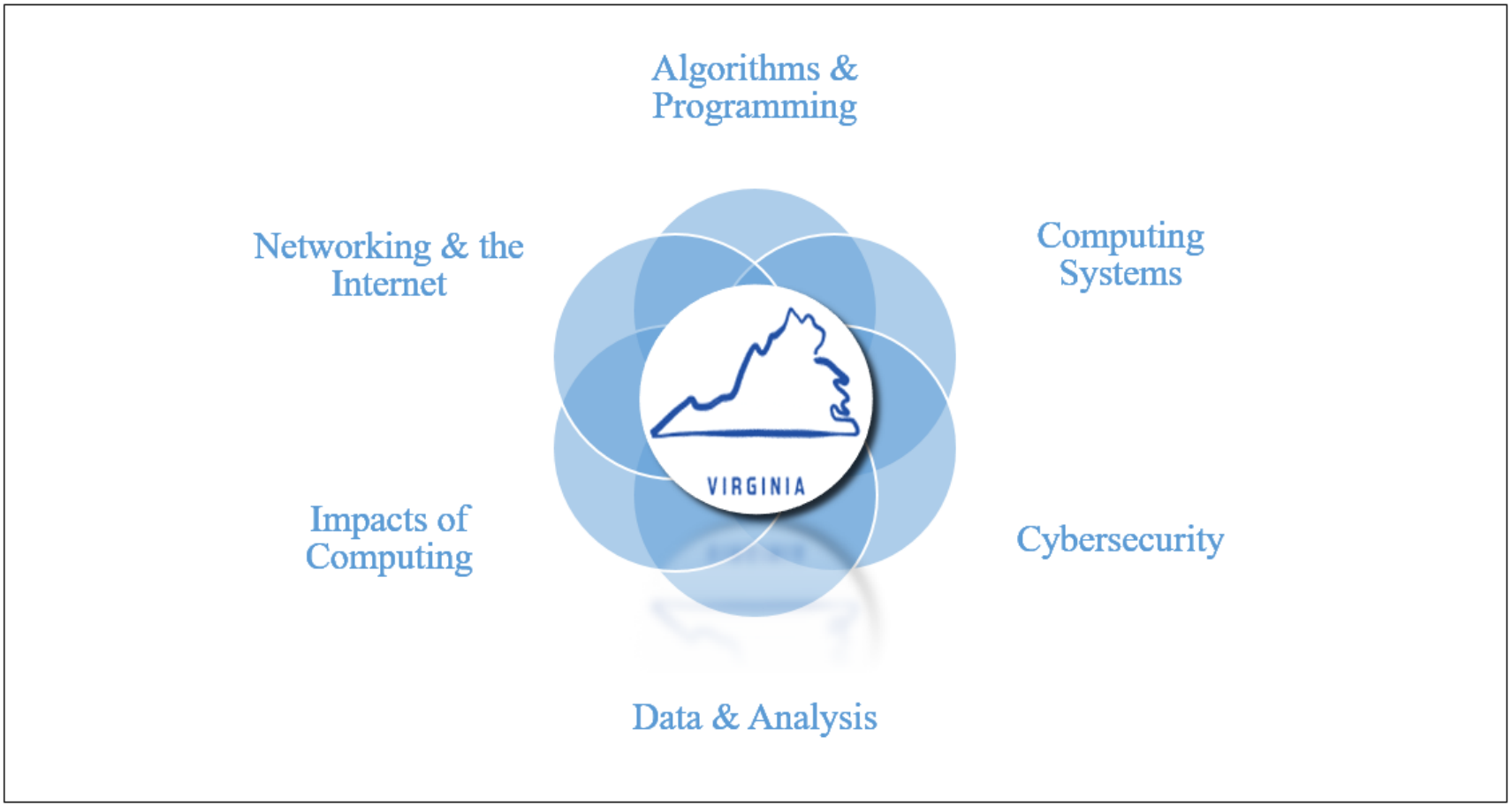
Computer Science Standards of Learning

Curriculum Framework



Board of Education
Commonwealth of Virginia

Copyright © 2019
by the
Virginia Department of Education
P.O. Box 2120
Richmond, Virginia 23218-2120
http://www.doe.virginia.gov

All rights reserved. Reproduction of these materials for instructional purposes in public school classrooms in Virginia is permitted.

Superintendent of Public Instruction
James F. Lane, Ed.D.

Assistant Superintendent of Learning
Gena Keller

Office of Science, Technology, Engineering, and Mathematics
Tina Manglicmot, Ed.D., Director
Anne Petersen, Ph.D., Science Coordinator
Timothy Ellis, Computer Science Specialist
Joshua Bearman, Science Specialist

NOTICE
The Virginia Department of Education does not discriminate on the basis of race, sex, color, national origin, religion, age, political affiliation, veteran status, or against otherwise qualified persons with disabilities in its programs and activities.

The 2017 *Computer Science Curriculum Framework* can be found on the Virginia Department of Education's Web site.

Introduction

The *Computer Science Standards of Learning* Curriculum Framework amplifies the *Computer Science Standards of Learning for Virginia Public Schools* and defines the content knowledge, skills, and understandings that are measured by the Standards of Learning. The Computer Science Curriculum Framework provides additional guidance to school divisions and their teachers as they develop an instructional program appropriate for their students. It assists teachers as they plan their lessons by identifying essential questions and vocabulary to drive instruction and defining the essential skills students should demonstrate. This supplemental framework delineates in greater specificity the minimum content that all teachers should teach and all students should learn.

School divisions should use the *Computer Science Curriculum Framework* as a resource for developing sound curricular and instructional programs. This framework should not limit the scope of instructional programs. Additional knowledge and skills that can enrich instruction and enhance students' understanding of the content identified in the Standards of Learning should be included as part of quality learning experiences.

Each topic in the *Computer Science Standards of Learning* Curriculum Framework is developed around the Standards of Learning. The format of the Curriculum Framework facilitates teacher planning by broadening the context of the standards and identifying essential student skills that should be the focus of instruction for each standard.

Context of the Standard

The Context of the Standard provides educators an explanation of the standard, including a description and the vertical development of the concept. This context will support teachers in incorporating computer science content into discipline-specific lessons. The intention of the Computer Science standards in grades K-8 is that Computer Science principles be integrated throughout content area instruction.

Essential Skills

The Essential Skills define student performance expectations aligned to each standard. The intent of the K-8 computer science standards is that the concepts are integrated into existing disciplines and this will result in these skills being emphasized differently in each content area. The expectation is that these Essential Skills are partnered with content area performance expectations as appropriate in instruction. At the high school level, the expectations in the 2017 *Computer Science Standards of Learning Curriculum Framework* are to be used in the support of standalone computer courses; the essential skills outlined in the document are not intended to be integrated into other coursework unless a teacher chooses to use the content to support discipline practices.

Essential Questions
Each standard has identified key questions to drive classroom instruction. These questions lead teachers and students toward the big ideas of each concept and provide a more holistic viewpoint used to lead instruction relating to the context of each standard.

Essential Vocabulary
In order to effectively communicate Computer Science concepts, essential vocabulary terms are defined in grade-level appropriate terms. These definitions are found in the glossary (Appendix A).

Middle School Computer Science Elective (MSCSE) Standards

The standards below outline the content for a flexible elective course with optional modules for 6-week, 9-week, 18-week, or 36-week implementations. These standards build on the concepts of computer science developed in prior grade levels and in the integrated standards for middle school students. Teachers are encouraged to select programming languages and environments, problems, challenges, and activities that are appropriate for their students to successfully meet the objectives of the standards.

The content for the initial 6-week module has an emphasis on computer programming. Students will review and build on skills developed throughout elementary school. Teachers may choose a block-based or text-based programming environment based on the prior experience of the students and the selected problems. For a 9-week module, students will study the history of computers and computer science, with a focus on the impact of Virginians. In the 18-week module, students will build additional programming skills within the framework of computer science principles. For an 18-week module, students will complete one or more projects to include programming, hardware and software integration, and collaboration.

Programmable computing tools will be used to facilitate design, analysis, and implementation of computer programs. Students for exploring and creating computer programs, facilitating reasoning and problem solving, and verifying solutions should use these tools.

6-week Core Module

Algorithms and Programming

MSCSE.1 The student will design and iteratively develop programs that combine control structures, including loops and conditionals.

Context of the Standard
Programs are collections of code organized in algorithms that can accomplish a variety of tasks. Programs can be developed to perform calculations, manipulate data, or simply to be creative. Programs can involve different control structures such as loops or if-statements and variables; these control structures are blocks of programming that analyzes variables within the program code to adjust and use accurate values as they change. Control structures help students develop their problem solving skills, and foster computational thinking. Effective variable use makes the problem solving process easier and faster.

Context of the Standard
In middle school, students continue their study of programming through a focus on algorithms. They work both collaboratively and individually to develop algorithms to reflect tasks in daily life; these algorithms become more complex as they recognize and use loops and events in the algorithms they construct.

Essential Skills	**Essential Questions**	**Essential Vocabulary**
Students should *demonstrate* these skills: • Develop, test, and revise a program using an iterative process. • Create a program that uses a combination of control structures. • Utilize a combination of mathematical and logical structures in a program. • Test a program for correctness. • Adjust and debug programs to be more efficient.	Students should *investigate* these concepts: • How does an iterative design process facilitate program development?	Students should *apply* these terms in context: • Control structure • Iterative process

MSCSE.2 The student will investigate variables and data types, including simple operations on strings.

Context of the Standard
Variables are programming elements capable of storing a piece of data. A variable is like a container with a name, in which the contents may change, but the name (identifier) does not. The identifier makes keeping track of the data that is stored easier,

Context of the Standard
especially if the data changes. Naming conventions for identifiers, and thoughtful choices of identifiers, improve program readability. Variables can hold a few different types of data: numbers, Boolean values (i.e. true and false), text characters, or strings of characters to make a word or phrase. Different data types will require the variable holding it to use different amounts of memory, and they operate differently from each other. A string is also a data type used in programming used to represent text rather than numbers. Quotations are used to represent data in a string. Operations can be conducted by using both data and strings. For example, in some environments adding two numbers together will produce the sum of their values, where two strings added will result in their values placed end-to-end ("cat" + "dog" = "catdog").

Essential Skills	**Essential Questions**	**Essential Vocabulary**
Students should *demonstrate* these skills: • Identify different types of data to be held in variables. • Conduct simple operations using non-numeric data types.	Students should *investigate* these concepts: • What types of values can a variable hold? • How may different data types interact when completing operations?	Students should *apply* these terms in context: • Variable • Strings

MSCSE.3 The student will implement a program that accepts input values, stores them in appropriately named variables, and produces output.

Context of the Standard
Computer systems are complex machines that primarily perform four main tasks: take input, store values, process data, and give output. Computer programs also typically follow a standard process called the input-process-output model. Data can be inputted into a program from the user, a data file, or a sensor. The processing stage can involve mathematical calculation, or logical control through the use of variables. These variables are created by the user and should be given names indicative of their content (e.g., a variable named person_height vs. var1). Once results are calculated, they should be produced for use. This output could be in the form of visual text, graphics, audio, or data values to be used by another program.

Essential Skills	**Essential Questions**	**Essential Vocabulary**
Students should *demonstrate* these skills: • Develop a program that takes assorted user input, assigns variables, and provides output that meets the expectations of an assigned task. • Create appropriately named variables to store values used in the program. • Produce output made from various stored variables.	Students should *investigate* these concepts: • What factors should you consider when naming variables? • What are examples of different outputs generated by an app?	Students should *apply* these terms in context: • Variable

MSCSE.4 The student will document programs in order to make them easier to trace, test, and debug.

Context of the Standard
When creating programs, it is highly recommended to use documentation and comments. Documentation allows programmers and reviewers to understand the intention behind a particular portion of code. Documentation comments are formatted to describe portions of the program but are not actually integrated into the program flow. Often, a programmer will write code and not return to it for lengths of time; documentation can be used to remind them of the intention behind their code and is very useful in debugging, as it signals how the program should be acting. The use of documentation when creating and redesigning a program greatly improves the efficiency of code review. Well-documented programs are easier to follow, and individual portions can have their function identified. If that function is what is malfunctioning in the program, debugging can be completed more quickly and easily.

Essential Skills	**Essential Questions**	**Essential Vocabulary**
Students should *demonstrate* these skills: • Use proper program documentation forms. • Use documentation to explain the purpose and function of a section of code.	Students should *investigate* these concepts: • How does program documentation make programming easier? • Why do comments make it easier to read a program written by someone else?	Students should *apply* these terms in context: • Documentation

Additional Content for 9-week Module
Impacts of Computing

MSCSE.5 The student will discuss issues of bias and accessibility in the design of existing technologies.

Context of the Standard
The construction of a program or innovation not only needs to focus on the parameters of that task or problem; it also needs to address the needs and backgrounds of the intended users. Bias and accessibility are two issues that must be taken into consideration when designing a program. Bias is defined as prejudice in favor of or against one thing, person, or group compared with another. Examples of this bias could be as obvious as a programmer creating characters in a game that only reflect particular genders or ethnicities, or more subtle, like the choice of language used in a program's output, or the choices made by a seemingly random program that actually favor a particular group or preference. Accessibility refers to the design of products, devices, services, or environments for people with disabilities. These disabilities may include visual, auditory, motor, and cognitive disabilities. The Web Content Accessibility Guidelines (WCAG) defines criteria to ensure that Web content is more accessible to people with disabilities.

Essential Skills	**Essential Questions**	**Essential Vocabulary**
Students should *demonstrate* these skills: • Identify a bias present in an existing technology. • Identify an accessibility issue in an existing technology. • Suggest a fix to an accessibility issue in a current technology.	Students should *investigate* these concepts: • What types of bias can be seen in a current technology that you use on a regular basis? • How do companies and programmers address accessibility concerns? • Why is it beneficial to remove bias from a technological application?	Students should *apply* these terms in context: • Bias • Accessibility

MSCSE.6 The student will describe and explain the history of computer science, including naming significant historical figures and describing their impact on the field.

Context of the Standard

The use of devices to aid in computation date back to ancient times with mechanical counting devices like the abacus. These devices grew in complexity, and had their purposes expand in function. Charles Babbage created the idea of a computer that could be programmed to accomplish a variety of tasks. Ada Lovelace extended that idea, and published the first "algorithm," or set of instructions to be carried out by a computer.

Alan Turing, a British World War II code-breaker, theorized that computers and algorithms could be broadened to accomplish any task. Grace Hopper was a programmer with the U.S. Navy who developed an early programming language called COBOL and coined the term "bug". As with many other accomplishments, the development of computers and programming languages is a collaborative effort that involves the work of teams. Over the next few decades, computers became smaller, more powerful, and more available to the public.

The Internet was created as a way to connect computers across great distances. A new system of organization on the Internet was created in 1991 by Sir Tim Berners-Lee called the World Wide Web. This was a set of rules that unified how Internet communication could happen on a worldwide scale. Programs on the web also became faster and more powerful over time. Now, computers have gotten very small but very powerful. They are embedded into more everyday devices to increase connectivity and functionality. Computer science has greatly shaped society, and computers will continue to become more integral in our everyday lives.

Essential Skills	Essential Questions	Essential Vocabulary
Students should *demonstrate* these skills: • Describe the function of early computational devices. • Create a timeline showing the major developments in computing technology.	Students should *investigate* these concepts: • How did the purpose of computers change over time? • How did advances in computing technology affect the pace of change in related fields (mathematics, science, engineering, etc.)? • Do you believe that advancement in computing was only possible because of the accomplishments of particular people, or would these advancements have happened regardless?	Students should *apply* these terms in context: • Computer

Additional Content for 18-week Module
Algorithms and Programming

MSCSE.7 The student will use flowcharts and/or pseudocode to address complex problems as algorithms.

Context of the Standard
In computer science, the development of programs uses an iterative design process involving design, implementation (programming), and review (debugging) until the program runs correctly. The design stage occurs before beginning to program. The planning stage is when the programmers gather information about the problem and sketch out a solution. This design process may include the use of pseudocode - writing out the steps of a program in English to make sure the flow of control and logic make sense. Flowcharts are another tool programmers may use when designing an algorithm or computer program. The flowchart outlines the steps that are needed in the development of an algorithm or program.

Essential Skills	Essential Questions	Essential Vocabulary
Students should *demonstrate* these skills: • Create a visualization of the path through a program.	Students should *investigate* these concepts: • Why should you plan a design for a program before starting to write it? • How are roads and traffic signs like a flowchart in programming? • How does pseudocode help programmers organize their thoughts?	Students should *apply* these terms in context: • Pseudocode

MSCSE.8 The student will incorporate existing code, media, and libraries into original programs, and give attribution.

Context of the Standard
As programs grow more complex, the choice of resources that aid program development becomes increasingly important. These resources include existing code, available media, and libraries. Repositories of sections of code are available for use online. Libraries are larger groupings of usable code that serve a particular purpose. Libraries contain collection of files, programs, routines, scripts, or functions that can be referenced in the programming code. They are often packaged up and distributed for easy inclusion in a program. Regardless of the source of a resource that may be used in the construction of a program, it is important that proper attribution is given to any borrowed components.

Essential Skills	Essential Questions	Essential Vocabulary
Students should *demonstrate* these skills: • Integrate portions of code, algorithms, and/or digital media from other sources in their own programs and websites. • Give attribution to the original creators to acknowledge their contributions.	Students should *investigate* these concepts: • Why should you give attribution to the original creator of a product? • How does the use of existing code or libraries accelerate program development?	Students should *apply* these terms in context: • Code library • Cite • Source • Plagiarism

MSCSE.9 The student will systematically test and refine programs using a range of test cases.

Context of the Standard
As part of the iterative design process, programs should be reviewed and tested to determine if the design goal is met through the generation of specific output. Testing programs using a range of test cases is a necessary step for assessing program correctness. A test case is a single input with an expected output to test the correctness of a program. Programmers need to test a program to make sure it works for all the possible values received as input. This can include values within the range of expected values as well as those outside the given range. Often, programmers will use "edge cases" - parameters that test the extremes of a scenario - to test their programs, ensuring that they will work with even the largest and smallest inputs. In beta test situations programmers use random people to test a program to find "bugs." Programmers record "versions" of programs as they update and improve their code.

Essential Skills	Essential Questions	Essential Vocabulary
Students should *demonstrate* these skills: • Predict the outcome or output of a program by examining written code. • Use multiple test cases to test the logic and mathematics of a program.	Students should *investigate* these concepts: • Why is there a need to debug? • How does program tracing help with the debugging process? • How can you predict the outcome of a program without running it?	Students should *apply* these terms in context: • Bug • Debug

Networks and the Internet

MSCSE.10 The student will model the role of protocols in transmitting data across networks and the Internet.

Context of the Standard
Protocols are sets of rules or procedures defined for particular actions. Protocols describe established commands and responses between computers on a network, such as requesting data or sending an image. Even simple messages sent over the Internet have a number of protocols used to facilitate the transmission of the message. These protocols work together and rely on each other to handle translation and transmission. There are many examples of protocols including TCP/IP (Transmission Control Protocol/Internet Protocol) and HTTP (Hypertext Transfer Protocol), which serve as the foundation for formatting and transmitting messages and data, including pages on the World Wide Web. Routers also implement protocols to record the fastest and most reliable paths by sending small packets as tests. *The student is not expected to know details of how specific protocols work in computing systems.*

Essential Skills	Essential Questions	Essential Vocabulary
Students should *demonstrate* these skills: • Identify the role of protocols in network communication. • Summarize how protocols make a decentralized Internet possible (i.e., no governing authority over the Internet).	Students should *investigate* these concepts: • How would the Internet function differently if there were no set protocols? • What are some examples of protocols that humans use to communicate with one another?	Students should *apply* these terms in context: • Protocol

Cybersecurity

MSCSE.11 The student will apply multiple methods of encryption to model the secure transmission of information.

Context of the Standard
In order to protect the security of data transmission on the Internet, encryption is used to hide sensitive data. There are tradeoffs between digital security and usability, speed, and cost. As needs and resources are considered, encryption techniques can be chosen appropriately. Encryption is an important aspect of communicating in a public system like the Internet. Without it, anybody would be able to view personal identifying information of anyone using the system from the packages that they send over the Internet. Some methods of encryption include Caesar cipher and Vigenere cipher. Caesar cipher uses an alphabetic shift to mask the true contents of a message. This is the easiest form of encryption to use and break. Vigenere uses a keyword to drive character substitution making the encryption more difficult to crack.

Essential Skills	Essential Questions	Essential Vocabulary
Students should *demonstrate* these skills:	Students should *investigate* these concepts:	Students should *apply* these terms in context:

Essential Skills	Essential Questions	Essential Vocabulary
• Identify different types of data that could be encrypted. • Use simple encryption strategies to encode a message.	• How might a security expert protect data on a network? • Why is encryption necessary when using computers for communication? • What types of data should be encrypted?	• Encryption • Data security

MSCSE.12 The student will explain how physical and digital security measures protect electronic information.

Context of the Standard
Students will extend their knowledge of the need for physical and digital security measures to understanding and explaining how both types of measures protect our data and personal information. Students can explore different types of security measures and understand how each one protects us in different ways from different types of electronic breaches and attacks. Physical security measures include locking rooms with devices, badges, fingerprints, security cameras, paper shredding, etc. Digital security measures include firewalls, anti-virus software, strong passwords, anti-spyware, etc.

Essential Skills	Essential Questions	Essential Vocabulary
Students should *demonstrate* these skills: • Identify examples of physical and digital security measures. • Explain why data security is necessary.	Students should *investigate* these concepts: • What are examples of physical security protections and how do they protect data? • What are examples of digital security protections and how do they protect data?	Students should *apply* these terms in context: • Password • Firewall

Essential Skills	Essential Questions	Essential Vocabulary
	• Why should a student implement security measures when working with digital information?	

Data and Analysis

MSCSE.13 The student will collect data using computational tools and transform the data to make it more useful and reliable.

Context of the Standard
Data collection is essential to help provide insight into patterns and other phenomena. These data can be collected by hand or through the use of computational tools such as sensors or surveys. Automated data collection (e.g., smartwatches regularly collect data on a person's heart rate, steps, etc.) is very efficient at gathering large sets to show patterns on a larger scale. The data often needs to be sorted, counted, and otherwise manipulated in order to make it easier to analyze. This manipulation can be done using spreadsheets, graphing, or other specialized software.

Essential Skills	Essential Questions	Essential Vocabulary
Students should *demonstrate* these skills: • Use a computational tool to collect a large set of data. • Transform data into different formats using computational tools.	Students should *investigate* these concepts: • Why is raw data typically difficult to use in analysis? • How can we collect data electronically? • What are examples of tools that can be used to collect data? • What are different ways data sets can be organized and displayed?	Students should *apply* these terms in context:

MSCSE.14 The student will refine computational models based on the data they have generated.

Context of the Standard
Once collected, data must be analyzed in order to make predictions and observations. These data can be used to build models that can replicate a phenomenon. Modeling means creating a physical replica or equations of a situation or activity. Programmers will create computational models to describe real-world systems. Computational models can be created in pseudocode (written) or equations, with robots, in block languages (i.e., Scratch), or in text-based languages (Java, Python, etc.). There are also content specific tools for modeling such as MatLab for mathematics. Some examples of computation models might be modeling and simulating a bungee jump, investigating the temperature of melting ice, or investigating the forces and attractions in the states of matter. Models are also constructed to test hypotheses. The data generated from the model are used to evaluate whether the models are accurate, to make adjustments, and draw conclusions. Models need to be tested for accuracy and refined as necessary. If a model generates an unexpected result based on data, then the model may need adjustment. Programmers engage in an iterative process as they analyze data and revise models multiple times in order to generate accurate models.

Essential Skills	**Essential Questions**	**Essential Vocabulary**
Students should *demonstrate* these skills: • Manipulate a data set in order to create a model. • Adjust a computational model to more accurately reflect a data set.	Students should *investigate* these concepts: • How can you tell if a model needs adjustment? • What components of a computational model can be adjusted? • What factors are considered when evaluating the accuracy of a model?	Students should *apply* these terms in context: • Data

MSCSE.15 The student will represent data using multiple encoding schemes.

Context of the Standard
When data is being stored on a computer, there are several options for how it will be organized. Some systems of organization are easier for computers to store and process data, while others can be efficiently read and analyzed by the user. Encoding schemes are systems of rules for converting data into a different format for a new purpose. Some schemes are simple and hold basic text but have limitations based on memory used per character (ASCII). Some hold a wider variety of characters including emojis, foreign language characters, and specialized symbols; this encoding scheme has a greater flexibility due to size (Unicode). Some encoding schemes are designed to hold a variety of data types but take up considerably more memory space (HTML). *Students are not expected to make manual conversions between data forms using encoding schemes.*

Essential Skills	**Essential Questions**	**Essential Vocabulary**
Students should *demonstrate* these skills: • Explain the need for changing the format of a set of data with an encoding scheme. • Describe how an encoding scheme can represent data in other forms.	Students should *investigate* these concepts: • Why do people transform data into multiple different formats? • How can a computer represent data in its numeric-only format? • What are examples of different numeric systems used by computers to represent non-numeric things?	Students should *apply* these terms in context: • Encoding scheme

Impacts of Computing

MSCSE.16 The student will compare tradeoffs associated with computing technologies that affect people's everyday activities and career options.

Context of the Standard
Computer use has permeated nearly every aspect of people's daily lives. This has brought many benefits to society, with increased levels of communication and access to information. There are many tradeoffs pertaining to consistent access to computing technology. These could include identity theft, cyberstalking, catfishing, fraudulent purchases, etc. Inadvertent data collection makes many people feel as though their privacy is being invaded, particularly since companies can track user data and sell it to advertisers or companies for analysis.

Essential Skills	**Essential Questions**	**Essential Vocabulary**
Students should *demonstrate* these skills: • Compare and contrast benefits and risks of everyday use of computing technology.	Students should *investigate* these concepts: • What computing technologies do you use every day? • What are the benefits of consistent access to technology? • What are the drawbacks to consistent access to technology?	Students should *apply* these terms in context:

MSCSE.17 The student will collaborate with many contributors through strategies such as crowdsourcing or surveys when creating a computational artifact or visualization.

Context of the Standard
When conducting data analysis, a larger set of data allows analysts to make generalizations about the situation/phenomena to which the data describes. Strategies such as surveys or crowdsourcing allow students to collect data from sources outside their work group. Often, outside opinions or ideas can help a creator or group break through a deadlock by providing a fresh perspective. By soliciting these ideas through crowdsourcing, a programming team can know that their program represents the views and preferences of a wide swath of people. *Teacher note: Reference division and school protocols prior to engaging in surveying or other crowdsourcing means with students.*

Essential Skills	**Essential Questions**	**Essential Vocabulary**
Students should *demonstrate* these skills: • Use a form or survey to collect data from a wide variety of people.	Students should *investigate* these concepts: • Why is it better to collect large sets of data when conducting analysis?	Students should *apply* these terms in context: • Crowdsourcing

MSCSE.18 The student will describe tradeoffs between allowing information to be public and keeping information private and secure.

Context of the Standard
The Fourth Amendment of the US Constitution protects every American's right to privacy. However, this amendment was written long before the advent of the Internet, where personal information is easily accessed and, in some cases, exploited. Modern tech companies, such as Facebook, Google, Amazon, and others, mine users' personal data and online habits to better advertise content

Context of the Standard
and products towards individuals. Students will explore what information they think should be publicly available. Students discuss the benefits and drawbacks to keeping information private when compared to public release.

Essential Skills	**Essential Questions**	**Essential Vocabulary**
Students should *demonstrate* these skills: • Identify and describe what information is appropriate, safe and responsible to share publicly. • Identify and describe what information is appropriate, safe and responsible to keep private.	Students should *investigate* these concepts: • What are the benefits of sharing personal information online? • What kind of personal information is regularly collected through technology use? • What value is your or your family's personal information to another person or business? • What information is appropriate to be public? Private? • Why should we allow certain information to be public vs. private? What are the consequences?	Students should *apply* these terms in context:

Computing Systems

MSCSE.19 The student will systematically identify and correct problems with computing devices and their components.

Context of the Standard
As with any system, there are times that a computer system does not work as intended. Although computing systems may vary, common troubleshooting strategies can be used on them, such as checking connections and power or swapping a working part in place of a potentially defective part. Since computing devices are composed of an interconnected system of hardware and software, troubleshooting strategies may need to address both. Students are expected to use accurate terminology to describe simple problems with computer hardware and software. Common troubleshooting strategies, such as checking that power is available, checking that physical and wireless connections are working, and clearing out the working memory by restarting programs or devices, are effective for many systems.

Essential Skills	**Essential Questions**	**Essential Vocabulary**
Students should *demonstrate* these skills: • Identify when a device or program is not working properly. • Perform simple troubleshooting tasks. • Apply troubleshooting strategies as needed in a classroom setting.	Students should *investigate* these concepts: • What are different troubleshooting tactics you should try if a program is not working? • Why is it important to be as specific as possible when you are describing a problem?	Students should *apply* these terms in context: • Troubleshooting

MSCSE.20 The student will explore the relationship between hardware and software using the Internet of Things.

Context of the Standard
An increasing number of devices have gained Internet connectivity. It is now possible to communicate with everyday objects like house lights, thermostats, or a garage door opener. These devices can also assess their surroundings using input components such as sensors (hardware), apply their programming logic (software), and relay output to an appropriate device to perform an action (hardware and software). They can also relay information to a user via an app who can decide on appropriate action.

Essential Skills	**Essential Questions**	**Essential Vocabulary**
Students should *demonstrate* these skills: • Describe what makes a device part of the "Internet of Things." • Describe the difference between smart and normal devices.	Students should *investigate* these concepts: • How does an Internet-connected device behave differently from a non-connected device? • What is the "Internet of Things?"	Students should *apply* these terms in context: • Internet of Things

36-week Module
Algorithms and Programming

MSCSE.21 The student will

a. work in a team to distribute tasks;
b. maintain a timeline; and
c. use iterative design to solve problems, including peer review and feedback.

Context of the Standard
Development of computational and digital artifacts (program, image, audio, etc.) is an iterative process and often a collaborative effort. Working well in a group requires students to practice strategies during the planning, writing, and review process such as delegation of responsibility, peer-review, revision of contributions, and assigning self-imposed deadlines. Due to the highly collaborative nature of programming projects, team members need to employ effective communication strategies. In order to avoid redundant work or gaps in the necessary code, groups may need to have open communication to help standardize the flow of the program. In order to develop a program, tasks can be broken down and created in modules by different groups of students, which requires clear and consistent communication between these groups. Additionally, the peer review fosters effective communication, and helps students see a variety of coding styles as well as identify common mistakes. The creation of appropriate and realistic timelines is difficult to many students, particularly when working in groups. Students may need teacher guidance and scaffolding in the timeline construction process as well as regular reminders to look back and revise timelines as needed throughout the process. These deadlines are used to construct a timeline that informs group participants of responsibilities and expectations when completing a project such as the development of a computational artifact.

Essential Skills	**Essential Questions**	**Essential Vocabulary**
Students should *demonstrate* these skills: • Work in teams to plan, design, and revise a program to complete an assigned task.	Students should *investigate* these concepts: • How does an iterative process help in the completion of a project?	Students should *apply* these terms in context: • Iterative design

Essential Skills	Essential Questions	Essential Vocabulary
• Construct a timeline that reflects the expectations and responsibilities of the members of a group.	• How does the use of a timeline facilitate group work?	

MSCSE.22 The student will decompose problems and subproblems into parts to facilitate the design, implementation, and review of programs.

Context of the Standard
Decomposition breaks problems down into smaller pieces (subproblem). A subprogram is a sequence of instructions whose execution is invoked from one or more remote locations in a program. Problem decomposition allows for complex tasks to be solved in manageable chunks. Once a program has been broken down, it enables different group members to work on different parts at the same time. Decomposition can occur in the planning and design stage, allowing for the separate subprograms to be more easily tackled, or when debugging a long and complex program. Students will practice this skill on programs they wrote and written by others.

Essential Skills	Essential Questions	Essential Vocabulary
Students should *demonstrate* these skills: • Breakdown problems into subproblems to facilitate creation of a program.	Students should *investigate* these concepts: • Why does breaking a problem down into smaller problems make the overall task easier?	Students should *apply* these terms in context: • Decompose • Subprogram

Essential Skills	Essential Questions	Essential Vocabulary
• Use decomposition in all parts of the program development process (i.e. plan, design, and review).	• How does decomposing a program into subprograms help programmers when debugging a program? • How can decomposition be applied throughout the development process? • Why would using subproblems in a program be thought of as a time saving measure?	

MSCSE.23 The student will create functions with parameters to organize code and make it easier to reuse.

Context of the Standard
Functions are like variables, except instead of storing data they store lines of code. Programmers use functions to help "chunk" large, self-contained sections of code to help make code more readable and to allow code to be used multiple times without having to re-write it. This increases code reusability by allowing an algorithm to be referenced and used wherever appropriate. Parameters are additional information that comes with the command call (e.g., squareRoot() vs. squareRoot(9)). These parameters provide the ability for a function to be used in a variety of situations.

Essential Skills	Essential Questions	Essential Vocabulary
Students should *demonstrate* these skills: • Create functions within a programming environment to accomplish a task.	Students should *investigate* these concepts: • How do parameters increase the capability of a function?	Students should *apply* these terms in context: • Function • Parameter

Essential Skills	Essential Questions	Essential Vocabulary
• Create functions with parameters to use in a variety of scenarios.		

Computing Systems

MSCSE.24 The student will recommend improvements to the design of computing devices, based on an analysis of how users interact with the devices.

Context of the Standard
Development of computing technology is an ongoing process. New technology components often introduce new software features. Based on feedback from the use of a new feature, developers will make adjustments for the next round of software updates. This feedback may be from user recommendations or from analysis of specific uses of data and functions. The use of feedback when designing programs is a practice that can be used in multiple settings. Students can make similar recommendations for improvements to apps they use on their phones.

Essential Skills	Essential Questions	Essential Vocabulary
Students should *demonstrate* these skills: • Analyze the design of a computing device for the purpose of making recommendations. • Make recommendations for improvements to a computing device.	Students should *investigate* these concepts: • What improvements can you make to a computing device? • What is the role of data analysis in determining changes in computing devices?	Students should *apply* these terms in context: • User interface

MSCSE.25 The student will design projects that combine hardware and software components to collect and exchange data.

Context of the Standard
Hardware can be used in tandem with software to facilitate data analysis in an efficient way. Sensors are hardware components designed to collect data that would otherwise be difficult to collect by hand. This could be by the type of the data or the amount of data to be collected.

Essential Skills	**Essential Questions**	**Essential Vocabulary**
Students should *demonstrate* these skills: • Plan or sketch ideas combining hardware and software components that collect and exchange data. • Select the hardware and software components for project designs by considering factors such as functionality, cost, size, speed, accessibility, and aesthetics.	Students should *investigate* these concepts: • How can hardware and software components be combined in the completion of s specific task or the completion of project? • Why is important to know what kind of data you want to collect before you begin to design a project? • What are the different kinds of data you can collect using hardware and software?	Students should *apply* these terms in context: • Sensors • Probeware