# Virginia Board of Education Agenda Item

**Agenda Item:** **Date:** July 27, 2017

| | | | |
|---|---|---|---|
| **Title** | First Review of the Proposed 2017 *Computer Science Standards of Learning* | | |
| **Presenter** | Mark R. Saunders, Director, Office of Technology and Virtual Learning, Division of Instruction | | |
| **E-mail** | Mark.Saunders@doe.virginia.gov | **Phone** | (804) 786-0307 |

**Purpose of Presentation:**
Action required by state or federal law or regulation.

**Previous Review or Action:**
Other. Specify below:
Date: February 23, 2017
**Action Requested:** The Board of Education received the Projected Timeline for the Development of the Computer Science Standards of Learning.
Action will be requested at a future meeting. Specify anticipated date below:
Date: October 26, 2017
Action: Final Review

**Alignment with Board of Education Goals: Please indicate (X) all that apply:**

| | |
|---|---|
| | Goal 1: Accountability for Student Learning |
| X | Goal 2: Rigorous Standards to Promote College and Career Readiness |
| | Goal 3: Expanded Opportunities to Learn |
| | Goal 4: Nurturing Young Learners |
| | Goal 5: Highly Qualified and Effective Educators |
| | Goal 6: Sound Policies for Student Success |
| | Goal 7: Safe and Secure Schools |
| | Other Priority or Initiative. Specify: |

**Background Information and Statutory Authority:**
Goal 2: The Board of Education's comprehensive plan calls for a review and revision of all Standards of Learning.

Code of Virginia, Section 22.1-253.13:1-B… "The Board shall seek to ensure that the Standards of Learning are consistent with a high-quality foundation educational program. The Standards of Learning shall include, but not be limited to, the basic skills of communication (listening, speaking, reading, and writing); computation and critical reasoning, including problem solving and decision making; proficiency in the use of computers and related technology; computer science and computational thinking, including computer coding; and the skills to manage personal finances and to make sound financial decisions."

HB831, a bill to amend and reenact § 22.1-253.13:1 of the Code of Virginia, relating to public elementary and secondary schools, was approved during the 2016 session of the General Assembly. The bill amended the Code of Virginia to state that the Standards of Learning shall include "computer science and computational thinking, including computer coding". Using an established process and criteria, the Department of Education planned a development of 2017 Computer Science Standards of Learning. The projected timeline for the development of Computer Science Standards of Learning during 2017 was received by the Board of Education in February 2017.

In accordance with the timeline, the Virginia Department of Education (VDOE) produced a draft of the Proposed 2017 Computer Science Standards of Learning documents found in Attachment A. During the review and revision process several actions were taken.

Public comment was received from stakeholders on the development of the Proposed 2017 *Computer Science Standards of Learning*. During the period of February 10-July 1, 2017 a total of 12 comments were received, including 7 from K-12 educators.

A steering committee was convened to review public comments and make recommendations for development of the standards. The steering committee was comprised of 8 classroom teachers, 1 Elementary Science Coordinator, 1 Content Integration Instructional Specialist, 1 Mathematics Curriculum Specialist, 1 CTE Curriculum Supervisor, 1 university professor, and 1 computer science leader from a non-profit organization that also led grade-band and content subgroups of a teacher review committee. VDOE staff from Instruction served in an advisory capacity. The steering committee met March 15, and 28; April 5-7, 19-21, 26; and May 31, 2017. Between the formal in-person meeting dates, the grade level bands met virtually to complete tasks and revisions.

A teacher review committee was convened to review public comments and make recommendations for the development of the standards. The teacher review committee was comprised of the 14 steering committee members and 16 additional individuals including; 13 classroom teachers, 1 Instructional Technology Resource Teacher, 1 STEM Coordinator, and 1 computer science career switcher. Teacher review committee members represented 7 out of 8 Superintendents' Regions: 1, 2, 3, 4, 5, and 6. VDOE staff from Instruction and Assessment served in an advisory capacity. The teacher review committee met May 8-10, 2017.

Two external committee meetings were convened. The first external committee meeting was held on May 11, 2017. Members of the following organizations were invited: Virginia Community College System, Virginia Department of Juvenile Justice, Virginia Society for Technology in Education, Virginia School Board Association, Virginia Association of School Business Officials, Virginia Association of School Superintendents, Virginia's Governor's Schools, Innovative High Schools, Virginia Association of Colleges of Teacher Education, Virginia Education Association, Virginia Association of Elementary School Principals, Virginia Association of Secondary School Principals, CodeVA, James Madison University, Virginia Commonwealth University, State Council of Higher Education for Virginia, Amazon, and Vicom Infinity, Inc.

During the first external committee meeting input and feedback was received from members of the following organizations: Virginia Community College System, Virginia Department of Juvenile Justice, Virginia Society for Technology in Education, Virginia School Board Association, Virginia Association of School Superintendents, Virginia Governor's Schools, CodeVA, James Madison University, and Virginia Commonwealth University.

The second external committee meeting was held on June 15, 2017. Members of the following organizations were invited: NASA, Capital One, Mobelux, Altria, Carmax, Corgibytes, IpponUSA, Solution Inc., WorldView, Oracle, RVATech, Amazon, Vicom Infinity, Inc., Virginia Community College System, and CodeVA. In addition, a former NASA employee and a computer science teacher from Hanover displayed an interest and consequently were invited to attend the meeting.

Input and feedback was received from members of the following organizations: Virginia Community College System, Capital One, Oracle, WorldView, CodeVA, Solutions Inc., IpponUSA, Corgibytes, Carmax, Altria, Mobelux. In addition, the former NASA employee and teacher from Hanover attended the meeting and offered feedback and input.

Support during the review process was received from the Instruction Division staff (Career and Technical Education, STEM, and Humanities and Early Childhood offices), including providing Technology and Virtual Learning staff with insights from work existing content area standards, relationship of proposed standards to existing content areas standards, working with content review committees, and providing technical reviews of proposed standards.

Using input received electronically and in person during meetings, along with support from VDOE staff, the steering committee developed a draft of the proposed standards.

The Computer Science Standards of Learning identify academic content for essential components of the computer science curriculum at different grade levels for Virginia's public schools. The standards are not intended to encompass the entire curriculum for a given grade level or course. School divisions are encouraged to incorporate the standards into a broader, locally-designed curriculum.

**Summary of Important Issues:**
The attached draft of the proposed Computer Science Standards of Learning includes the following components, as listed.

Proposed 2017 Computer Science Standards of Learning

- Preface
- Introduction
- What is Computer Science?
- What is Computational Thinking
- Computer Science Practices for Students
- Equity
- Organization of the Computer Science Standards of Learning
- Standards of Learning by Grade Level and Content Strand

**Summary of Committee Expectations and Recommendations:**

During the steering committee and review committee meetings, members were asked to:

- review national resources in computer science education including
  - K-12 Computer Science Framework;
  - Computer Science Teachers Association Standards
  - Existing state computer science or related standards from the following states:
    - Arkansas

    - Indiana
    - Massachusetts
    - Washington
- review comments received through the official public comment period;
- review the Profile of a Virginia Graduate;
- review existing content area standards, including computer technology;
- recommend standards that
  - show the vertical progression of computer science content and skills
  - correlate to existing content area standards, including computer technology;
  - provide local school division flexibility in implementation;
  - ensure developmental appropriateness of student expectations;
  - sufficiently prepare students for college and careers

**Impact on Fiscal and Human Resources:**
The collection and analysis of public comment and additional revisions to the standards and Curriculum Framework can be absorbed by the agency's existing resources at this time. If the agency is required to absorb additional responsibilities related to this activity, other services may be impacted.

**Timetable for Further Review/Action:**
Following the Board of Education's acceptance of the Proposed 2017 Computer Science Standards of Learning for first review, the Department of Education will receive public comment for at least 30 days before bringing the Proposed 2017 Computer Science Standards of Learning to the Board of Education for final review in October 2017. During the public comment period, the Board of Education will host public hearings on the Proposed 2017 Computer Science Standards of Learning.

**Superintendent's Recommendation:**
The Superintendent of Public Instruction recommends that the Board of Education receive the Proposed 2017 Computer Science Standards of Learning for first review.

Attachment A

# Proposed Computer Science Standards of Learning

## For Virginia Public Schools

**Board of Education**
**Commonwealth of Virginia**

**July 2017**

# Proposed Computer Science Standards of Learning

## For Virginia Public Schools

**Reviewed in July 2017 by the Board of Education**
Billy K. Cannaday, Jr., Ed.D., President
Daniel A. Gecker, Vice President
Kim E. Adkins
Diane T. Atkinson
James H. Dillard
Anne B. Holton
Elizabeth V. Lodal
Sal Romero, Jr.
Jamelle S. Wilson, Ed.D.

**Superintendent of Public Instruction**
Steven R. Staples, Ed.D.

Commonwealth of Virginia
Board of Education
Post Office Box 2120
Richmond, VA 23218-2120

© July 2017

Copyright © 2017
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**
Steven R. Staples, Ed.D.

**Chief Academic Officer/Assistant Superintendent for Instruction**
Steven M. Constantino, Ed.D.

**Office of Technology and Virtual Learning**
Meg Foley, Special Projects Coordinator
Jean B. Weller, Educational Technology Specialist

**Acknowledgements**
The Virginia Department of Education wishes to express sincere thanks to Mark R. Saunders, who assisted in the development of the 2017 *Computer Science Standards of Learning*.

**Notice to Reader**
In accordance with the requirements of the Civil Rights Act and other federal and state laws and regulations, this document has been reviewed to ensure that it does not reflect stereotypes based on race, color, national origin, sex, age, or disability.

The Virginia Department of Education does not discriminate on the basis of race, color, national origin, sex, age, or disability in employment or provisions of service.

# Preface

In 1995, the Virginia Board of Education published Standards of Learning in English, mathematics, science, and history and social science for kindergarten through grade 12. Subsequently, Standards of Learning were developed for all academic content areas. The Standards of Learning provide a framework for instructional programs designed to raise the academic achievement of all students in Virginia and prepare them for college and careers.

The Standards of Learning are recognized as a model for other states. Pursuant to legislation from the 2000 Virginia General Assembly, the Board of Education established a seven-year cycle for review of the Standards of Learning. The 2016 Virginia General Assembly approved legislation requiring that the Standard of Learning include computer science and computational thinking, including computer coding. As a result, the Computer Science Standards of Learning were developed in 2017. The standards were developed with input from parents, teachers, administrators, representatives from higher education, and the business community. The standards set clear, concise, and measurable academic expectations for students. Parents and guardians are encouraged to work with their children, their children's teachers, and their children's schools to help them achieve these academic standards.

# Introduction

The 2017 Computer Science Standards of Learning for Virginia Public Schools identify academic content for essential components of the computer science curriculum at different grade levels. Information from Computer Science Teachers Association (https://www.csteachers.org/), K-12 Computer Science Framework (https://k12cs.org/), College Board Advanced Placement Computer Science courses (http://apcentral.collegeboard.com/home ), Exploring Computer Science (http://www.exploringcs.org/), were considered in identifying computer science content necessary for success for all students in postsecondary pursuits.

Standards are identified for kindergarten through grade eight, with an optional selection of electives modules at the middle school level, and a sequence of high school courses. The standards are organized into the following content strands: Computing Systems, Networks and the Internet, Cybersecurity, Data and Analysis, Algorithms and Programming, and Impacts of Computing. The Standards of Learning within each strand progress in complexity throughout the grade levels and into high school course content. While the standards are organized by strand and identified numerically, local curricula and pacing guides should determine the instructional sequence of the content.

The K - 8 standards were designed to be integrated into instruction in multiple subject areas including mathematics, science, history, English, fine arts, and career and technology courses. The middle school and high school electives are separate courses and modules, but where appropriate, connections are made to content in other disciplines. The high school standards are designed to provide flexibility in application of core ideas to various contexts.

The core practices of computer science, including collaboration, communication, and fostering an inclusive culture, describe the behaviors and ways of thinking that computationally literate students use to fully engage in today's data-rich and interconnected world. Collaborative computing is the process of performing a computational task by working in pairs and on teams, including working with individuals with diverse perspectives, skills, and personalities. Students will need to solicit and incorporate the feedback of others, which can lead to better outcomes than working independently. Students should use collaborative tools to effectively work together and to create complex artifacts. In computer science, students communicate with diverse audiences about the use and effects of computation and the appropriateness of computational choices. Students describe, justify, and document computational processes through multiple forms of media and give appropriate attribution. Clear communication includes using precise language and carefully considering possible audiences.

The use of appropriate technology and the interpretation of the results from applying technology tools must be an integral part of teaching, learning, and assessment. While some computational thinking practices can be developed with or without technology, the use of computing devices is essential for students to meet the objectives of these standards.

The Virginia Department of Education strongly encourages educators to make equitable access a guiding principle in the implementation of these computer science standards by giving all students the opportunity to learn computer science. We encourage the elimination of barriers that restrict access to computer science for students from ethnic, racial and socioeconomic groups that have traditionally been underrepresented. Schools should make every effort to ensure their computer science classes reflect the diversity of their student population. It is only through a commitment to equitable preparation and access that true equity and excellence can be achieved.

## What is Computer Science?

The Computer Science standards integrate computer literacy, educational technology, digital citizenship, information technology, and computer science. Computer literacy, educational technology, digital citizenship, and information technology are concepts that students are also exposed to in the Computer Technology Standards of Learning. In many ways, instruction in the Computer Science standards will compliment and expound upon, at a deeper level, the concepts and skills covered with the Computer Technology standards. However, there are distinct differences between computer technology and computer science. As the foundation for all computing, *computer science* is "the study of computers and algorithmic processes, including their principles, their hardware and software designs, their applications, and their impact on society" (Tucker et. al, 2006, p. 2).

Computer science builds upon the concepts of computer literacy, educational technology, digital citizenship, and information technology. The differences and relationship with computer science are described below.

- *Computer literacy* refers to the general use of computers and programs, such as productivity software. Examples include performing an Internet search and creating a digital presentation.

- *Educational technology* applies computer literacy to school subjects. For example, students in an English class can use a web-based application to collaboratively create, edit, and store an essay online.

- *Digital citizenship* refers to the appropriate and responsible use of technology, such as choosing an appropriate password and keeping it secure.

- *Information technology* often overlaps with computer science but is mainly focused on industrial applications of computer science, such as installing software rather than creating it. Information technology professionals often have a background in computer science.

## What is Computational Thinking?

Also integrated throughout the Computer Science standards is the concept of computational thinking. *Computational thinking* is an approach to solving problems in a way that can be implemented with a computer. It involves the use of concepts, such as abstraction, recursion, and iteration, to process and analyze data, and to create real and virtual artifacts [Computer Science Teachers Association & Association for Computing Machinery]. Computational thinking practices such as abstraction, modeling, and decomposition connect with computer science concepts such as algorithms, automation, and data visualization. Beginning with the elementary school grades and continuing through grade 12, students should develop a foundation of computer science knowledge and learn new approaches to problem solving that captures the power of computational thinking to become both users and creators of computing technology.

## Computer Science Practices for Students

The content of the Computer Science standards is intended to support the following seven practices for students: fostering an inclusive computing culture, collaborating around computing, recognizing and defining computational problems, developing and using abstractions, creating computational artifacts, testing and refining computational artifacts, and communicating about computing. The practices describe the behaviors and ways of thinking that computationally literate students use to fully engage in a data-rich and interconnected world. Computational thinking refers to the thought processes involved in expressing solutions as computational steps or algorithms that can be carried out by a computer (Cuny, Snyder, & Wing, 2010; Aho, 2011; Lee, 2016).

**Fostering an Inclusive Computing Culture**
Students will develop skills for building an inclusive and diverse computing culture, which requires strategies for incorporating perspectives from people of different genders, ethnicities, and abilities. Incorporating these perspectives involves understanding the personal, ethical, social, economic, and cultural contexts in which people operate. Considering the needs of diverse users during the design process is essential to producing inclusive computational products.

**Collaborating Around Computing**
Students will develop skills for collaborating around computing. Collaborative computing is the process of performing a computational task by working in pairs and on teams. Collaborative computing involves asking for the contributions and feedback of others, effective collaboration can lead to better outcomes than working independently. Collaboration requires individuals to navigate and incorporate diverse perspectives, conflicting ideas, disparate skills, and distinct personalities. Students should use collaborative tools to effectively work together and to create complex artifacts.

**Recognizing and Defining Computational Problems**
Students will develop skills for recognizing and defining computational problems. The ability to recognize appropriate and worthwhile opportunities to apply computation is a skill that develops over time and is central to computing. Solving a problem with a computational approach requires defining the problem, breaking it down into parts, and evaluating each part to determine whether a computational solution is appropriate.

**Developing and Using Abstractions**
Students will develop skills for developing and using abstractions. Identifying patterns and extracting common features from specific examples to create generalizations form abstractions. Using generalized solutions and parts of solutions designed for broad reuse simplifies the development process by managing complexity.

**Creating Computational Artifacts**
Students will develop skills for creating computational artifacts. The process of developing computational artifacts embraces both creative expression and the exploration of ideas to create prototypes and solve computational problems. Students create artifacts that are personally relevant or beneficial to their community and beyond. Computational artifacts can be created by combining and modifying existing artifacts or by developing new artifacts. Examples of computational artifacts include programs, simulations, visualizations, digital animations, robotic systems, and apps.

**Testing and Refining Computational Artifacts**
Students will develop skills for testing and refining computational artifacts. Testing and refinement is the deliberate and iterative process of improving a computational artifact. This process includes debugging (identifying and fixing errors) and comparing actual outcomes to intended outcomes. Students also

respond to the changing needs and expectations of end users and improve the performance, reliability, usability, and accessibility of artifacts.

**Communicating About Computing**
Students will develop skills for communicating about computing. Communication involves personal expression and exchanging ideas with others. In computer science, students communicate with diverse audiences about the use and effects of computation and the appropriateness of computational choices. Students write clear comments, document their work, and communicate their ideas through multiple forms of media. Clear communication includes using precise language and carefully considering possible audiences.

## Equity

The Computer Science standards were developed with the intent of equity, in the form of computer science for all, as a core belief. While equity was a core belief in the development of the Computer Science standards, additional effort at the local and state level with policies and at the school level with curriculum, instruction, and classroom culture is also necessary for equity to play a role in computer science education. The intent of equity in computer science is not to prepare all students to major in computer science at the higher education level and then pursue careers in software development or database management. The intent of equity is to ensure that all students have the basic knowledge that will allow them to productively participate in the world and make well informed decisions about their lives. Equity is not limited to whether classes are available, but also includes how classes are taught, how students are recruited for classes or activities, and how the classroom culture supports diverse learners and promotes the retention of students. The result of equity can be a diverse classroom of students, based on factors such as race, gender, disability, socioeconomic status, and English language proficiency, all of whom have high expectations and feel capable of learning.

Equity has also been considered not only in the composition of the groups writing, advising, and reviewing the Computer Science standards but also in the content of the standards by designing standards that can be engaged in by all students and are flexible enough to allow all students to demonstrate proficiency in multiple ways. The intent of allowing for multiple ways of demonstrating proficiency is further completed by the work of the Virginia Board of Education with the Profile of a Virginia Graduate and the use of locally-developed performance-based assessments.

## Organization of the Computer Science Standards of Learning

The Computer Science standards for kindergarten through grade twelve are organized into six strands: Algorithms and Programming, Computing Systems, Cybersecurity, Data and Analysis, Impacts of Computing, and Networks and the Internet. Each grade level is preceded by an overview that describes the major concepts and skills that each student will be expected to understand and demonstrate. The vertically aligned standards are intended to reflect a comprehensive instructional program and document a progression of expected achievement in each of the strands. This organization of standards also reflects the gradual progression in the development of skills.

Algorithms and Programming involves the use of algorithms. An algorithm is a sequence of steps designed to accomplish a specific task. Algorithms are translated into programs, or code, to provide instructions for computing devices. Algorithms and programming control all computing systems, empowering people to communicate with the world in new ways and solve compelling problems. The development process to create meaningful and efficient programs involves choosing which information to use and how to process and store it, breaking apart large problems into smaller ones, recombining existing solutions, and analyzing different solutions.

Computing Systems involves the interaction that people have with a wide variety of computing devices that collect, store, analyze, and act upon information in ways that can affect human capabilities both positively and negatively. The physical components (hardware) and instructions (software) that make up a computing system communicate and process information in digital form. An understanding of hardware and software is useful when troubleshooting a computing system that does not work as intended.

Cybersecurity, also known as information technology security, involves the protection of computers, networks, programs, and data from unauthorized or unintentional access, manipulation, or destruction. Many organizations, such as government, military, corporations, financial institutions, hospitals, and others collect, process, and store significant amounts of data on computing devices. That data is transmitted across multiple networks to other computing devices. The confidential nature of government, financial, and other types of data requires continual monitoring and protection for the sake of continued operation of vital systems and national security.

Data and Analysis involves the data that exist and the computing systems that exist to process that data. The amount of digital data generated in the world is rapidly expanding, so the need to process data effectively is increasingly important. Data is collected and stored so that it can be analyzed to better understand the world and make more accurate predictions.

Impacts of Computing involves the affect that computing has on daily life. Computing affects many aspects of the world in both positive and negative ways at local, national, and global levels. Individuals and communities influence computing through their behaviors and cultural and social interactions, and in turn, computing influences new cultural practices. An informed and responsible person should understand the social implications of the digital world, including equity and access to computing.

Network and the Internet involves the networks that connect computing systems. Computing devices do not operate in isolation. Networks connect computing devices to share information and resources and are an increasingly integral part of computing. Networks and communication systems provide greater connectivity in the computing world by providing fast, secure communication and facilitating innovation.

# Kindergarten

The kindergarten standards place emphasis on developing awareness of computing and computing devices by gathering and organizing data, by sorting or step-by-step. Students will use accurate terminology to identify components and describe their purposes. Students will also be introduced to communication, security, and responsible computing behaviors. The use of technology will be an integral part of successful acquisition of skills in all content areas.

## Algorithms and Programming

K.1 The student will construct algorithms (set of step-by-step instructions) either independently or collaboratively including sequencing, emphasizing the beginning, middle, and end. [Related SOL: Math K.13]

K.2 The student will construct programs to accomplish tasks as a means of creative expression using a block based programming language or unplugged activities, either independently or collaboratively, including sequencing, emphasizing the beginning, middle, and end.

K.3 The student will create a design document to illustrate thoughts, ideas, and stories in a sequential (step-by-step) manner (e.g., story map, storyboard, and sequential graphic organizer).

K.4 The student will categorize a group of items based on one attribute or the action of each item, with or without a computing device. [Related SOL: Math K.12, Science K.1d]

## Computing Systems

K.5 The student will identify components of computing systems (e.g., keyboard, mouse, desktop computer, laptop computer, tablet, and printer).

K.6 The student will identify, using accurate terminology, simple hardware and software problems that may occur during use (e.g., app or program is not working as expected, no sound is coming from a device, device will not turn on).

## Cybersecurity

K.7 The student will identify what is allowed and what is not allowed at school when using technology.

K.8 The student will identify personal information (e.g., address, telephone number, and name) and the importance of protecting personal information online.

## Data and Analysis

K.9 The student will gather and display data and organize it in a chart or graph in order to answer questions about the data, with or without a computing device. [Related SOL: HSS K.1c]

## Impacts of Computing

K.10 The student will identify responsible behaviors associated with using information and technology. [Related SOL: HSS K.10]

## Networking and the Internet

K.11 The student will discuss, in a whole class setting, how information can be communicated electronically.

# Grade One

The first grade standards place emphasis on developing organizational skills, such as classifying based on common attributes, completing a pattern, or explaining processes step-by-step. Students will use accurate terminology to identify components and describe their purposes. Students will also be able to describe to communication, security, and responsible computing behaviors. The use of technology will be an integral part of successful acquisition of skills in all content areas.

## Algorithms and Programming

1.1 The student will construct algorithms (set of step-by-step instructions) either independently or collaboratively, including
   a) sequencing (including ordinal numbers) and;
   b) simple loops (patterns and repetition).

1.2 The student will construct programs to accomplish tasks as a means of creative expression using a block based programming language or unplugged activities, either independently or collaboratively including
   a) sequencing, ordinal numbers; and
   b) simple loops (patterns and repetition).

1.3 The student will analyze and debug (correct and improve) an algorithm that includes sequencing.

1.4 The student will plan and create a design document to illustrate thoughts, ideas, and stories in a sequential (step-by-step) manner (e.g., story map, storyboard, sequential graphic organizer).

1.5 The student will categorize a group of items based on one or two attributes or the actions of each item, with or without a computing device. [Related SOL: Science 1.1c, Math 1.13]

1.6 The student will acknowledge that materials that are created by others (e.g., author, illustrator).

## Computing Systems

1.7 The student will identify components of computing systems that are common among different types of computing devices including desktop and laptop computers, tablets, and mobile phones.

1.8 The student will identify, using accurate terminology, simple hardware and software problems that may occur during use (e.g., app or program is not working as expected, no sound is coming from the device, the device won't turn on).

## Cybersecurity

1.9 The student will describe what is allowed and what is not allowed at school associated with the use of technology.

1.10 The student will identify and use strong passwords, explain why strong passwords should be used (e.g., protect name, address, and telephone number).

## Data and Analysis

1.11 The student will identify and interpret data and organize it in a chart or graph in order to make a prediction, with or without a computing device. [Related SOL: HSS 1.1c]

## Impacts of Computing

1.12 The student will identify and explain responsible behaviors associated with using information and technology. [Related SOL: HSS 1.10]

## Networking and the Internet

1.13 The student will, in a whole class environment, discuss how information can be communicated electronically (e.g., email, social media).

# Grade Two

The standards for second grade place an emphasis on creating models of physical objects or processes to demonstrate relationships. Second grade standards build on students' skills in constructing programs and utilizing algorithms. The accurate use of terminology as well as the responsible use of technology will continue to be built upon. The foundational understanding of computing and the use of technology will be an integral component of successful acquisition of skills across content areas.

## Algorithms and Programming

2.1 The student will construct algorithms (set of step-by-step instructions) both independently and collaboratively
- a) using sequencing;
- b) using loops (a wide variety of patterns such as repeating patterns or growing patterns); and
- c) identifying events.

2.2 The student will construct programs to accomplish tasks as a means of creative expression using a block based programming language or unplugged activities, both independently and collaboratively
- a) using sequencing;
- b) using loops (a wide variety of patterns, such as repeating patterns or growing patterns); and
- c) identifying events.

2.3 The student will analyze and debug (correct and improve) an algorithm that includes sequencing and simple loops, with or without a computing device.

2.4 The student will plan and create a design document to illustrate thoughts, ideas, and stories in a sequential (step-by-step) manner (e.g., story map, storyboard, sequential graphic organizer).

2.5 The student will compare and contrast a group of items based on the attributes or actions of each item, with or without a computing device. [Related SOL: Science.2.1d, Math 2.13]

2.6 The student will acknowledge that materials are created by others (e.g., author, illustrator, and website).

## Computing Systems

2.7 The student will describe the characteristics of computing systems to include hardware, software, input, and output.

2.8 The student will identify, using accurate terminology, simple hardware and software problems that may occur during use (e.g., app or program not working as expected, no sound, device won't turn on).

## Cybersecurity

2.9 The student will explain what is allowed and what is not allowed at school associated with the use of technology (e.g., class rules).

2.10 The student will identify and create strong passwords, explain why strong passwords should be used. (e.g., protect name, address, and telephone number).

## Data and Analysis

2.11 The student will construct and analyze data and organize it in a chart or graph in order to make a prediction, with or without a computing device. [Related SOL: HSS 2.1c]

2.12 The student will create a model of a physical object or process in order to show relationships with or without a computing device (e.g., water cycle, butterfly life cycle, seasonal weather patterns).

## Impacts of Computing

2.13 The student will compare and contrast examples of how computing technology has changed and improved the way people live, work, and interact. [Related SOL: HSS 2.11]

2.14 The student will identify and model responsible behaviors when using information and technology. [Related SOL: HSS 2.11]

## Networking and the Internet

2.15 The student will discuss as a class how information can be communicated electronically (e.g., email, social media, video conferencing, blogging).

# Grade Three

The standards for third grade place an emphasis on decomposing larger problems and utilizing the iterative design process to develop a plan to construct and execute programs. Students in third grade are introduced to using computing systems to model attributes and behaviors associated with a concept. The accurate use of terminology as well as the responsible use of technology will continue to be built upon. The foundational understanding of computing and the use of technology will be an integral component of successful acquisition of skills across content areas.

## Algorithms and Programming

3.1 The student will construct algorithms (set of step-by-step instructions) both independently and collaboratively

a) using sequencing;
b) using loops (a wide variety of patterns such as repeating patterns or growing patterns); and [Related SOL: Math 3.16]
c) using events.

3.2 The student will construct programs to accomplish tasks as a means of creative expression using a block or text based programming language, both independently and collaboratively a) using sequencing;

b) using loops (a wide variety of patterns such as repeating patterns or growing patterns); and [Related SOL: Math 3.16]
c) using events.

3.3 The student will analyze and debug (correct and improve) an algorithm that includes sequencing, events, and loops. [Related SOL areas – Math: Problem Solving, English: Editing]

3.4 The student will create a plan as part of the iterative design process, independently and/or collaboratively using strategies such as pair programming (e.g., storyboard, flowchart, pseudo-code, story map. [Related SOL: English 3.8c]

3.5 The student will compare and contrast a group of items based on attributes or actions classified into at least two sets and two subsets. [Related SOL: Science 3.1c]

3.6 The student will decompose (break down) a larger problem into smaller sub-problems, independently or collaboratively. [Related SOL: Math 3.3b]

3.7 The student will give credit to sources when borrowing or changing ideas (e.g., using information and pictures created by others, using music created by others, remixing programming projects). [Related SOL: English 3.10e]

## Computing Systems

3.8 The student will model how a computing system works including input and output; and [Related SOL: Math 3.16]

3.9 The student will identify, using accurate terminology, simple hardware and software problems that may occur during use, and apply strategies for solving problems (e.g., rebooting the device, checking for power, checking network availability, closing and reopening an app).

## Cybersecurity

3.10 The student will identify problems that relate to inappropriate use of computing devices and networks.

3.11 The student will create examples of strong passwords, explain why strong passwords should be used, and demonstrate proper use and protection of personal passwords.

## Data and Analysis

3.12 The student will answer questions by using a computer to observe data in order for the student to draw conclusions and make predictions. [Related SOL: Math 3.15, HSS 3.1d]

3.13 The student will create an artifact using computing systems to model the attributes and behaviors associated with a concept (e.g., day and night, animal life cycles, plant life cycles). [Related SOL areas – Math: Models, Science: Moon Phases]

## Impacts of Computing

3.14 The student will identify computing technologies that have changed the world and express how those technologies influence, and are influenced by, cultural practices.

3.15 The student will identify the positive and negative impacts of the pervasiveness of computers and computing in daily life (e.g., downloading videos and audio files, electronic appliances, wireless Internet, mobile computing devices, GPS systems, wearable computing).

3.16 The student will identify social and ethical issues that relate to computing devices and networks. [Related SOL: C/T: 6-8.3, HSS 3.11]

## Networking and the Internet

3.17 The student will identify as a class that information can be transmitted using computing devices via a network (e.g., email, blogging, video messaging).

# Grade Four

The fourth-grade standards place emphasis on constructing programs and utilizing algorithms to accomplish a task. Students continue to decompose larger problems into smaller tasks. In fourth grade, students begin to think about the impacts of computing and computing devices. The accurate use of terminology as well as the responsible use of technology will continue to be built upon. The foundational understanding of computing and the use of technology will be an integral component of successful acquisition of skills across content areas.

## Algorithms and Programming

4.1 The student will construct algorithms (set of step-by-step instructions) both independently and collaboratively

a) using sequencing;
b) using loops;
c) using variables to store and process data; and
d) performing number calculations on variables (e.g., addition, subtraction, multiplication and division). [Related SOL: Math 4.4, 4.5, 4.6]

4.2 The student will construct programs to accomplish a task as a means of creative expression using a block or text based programming language, both independently and collaboratively

a) using sequencing;
b) using loops;
c) using variables; and
d) performing number calculations (e.g., addition, subtraction, multiplication and division) on variables. [Related SOL: Math 4.4, 4.5, 4.6]

4.3 The student will analyze and debug (correct and improve) an algorithm that includes sequencing, events, loops and variables. [Related SOL areas – Math: Problem Solving, English: Editing]

4.4 The student will create a plan as part of the iterative design process, both independently and collaboratively using strategies such as pair programming (e.g., storyboard, flowchart, pseudo-code, story map). [Related SOL: English: 4.7d, f]

4.5 The student will classify and arrange a group of items based on the attributes or actions. [Related SOL: Science 4.1.b]

4.6 The student will decompose (break down) a larger problem into smaller sub-problems, both independently and collaboratively. [Related SOL: Math 4.4d]

4.7 The student will give credit to sources when borrowing or changing ideas (e.g., using information, pictures created by others, using music created by others, remixing programming projects). [Related SOL: English 4.9d, e]

## Computing Systems

4.8 The student will model how a computing system works including input and output, processors and sensors.

4.9 The student will identify, using accurate terminology, simple hardware and software problems that may occur during use, and apply strategies for solving problems (e.g.,

rebooting the device, checking for power, checking for network availability, closing and reopening an app).

## Cybersecurity

4.10 The student will identify and explain problems that relate to inappropriate use of computing devices and networks.

4.11 The student will create examples of strong passwords, explain why strong passwords should be used, and demonstrate proper use and protection of personal passwords.

## Data and Analysis

4.12 The student will answer questions by using a computer to manipulate data in order for the student to draw conclusions and make predictions. [Related SOL: Math 4.14]

4.13 The student will create an artifact using computing systems to model the attributes and behaviors associated with a concept (e.g., solar system). [Related SOL area – Math: Models]

4.14 The student will use numeric values to represent non-numeric ideas in the computer (binary, ASCII, pixel attributes such as RGB). [Related SOL: Math 5.19a]

## Impacts of Computing

4.15 The student will give examples of computing technologies that have changed the world and express how those technologies influence, and are influenced by, cultural practices.

4.16 The student will describe the positive and negative impacts of the pervasiveness of computers and computing in daily life (e.g., downloading videos and audio files, electronic appliances, wireless Internet, mobile computing devices, GPS systems, wearable computing).

4.17 The student will describe social and ethical issues that relate to computing devices and networks. [Related SOL: C/T: 6-8.3]

## Impacts of Computing

4.18 The student will identify and explain how information can be transmitted using computing devices via a network (e.g., email, images, and videos).

# Grade Five

The fifth-grade standards place emphasis on constructing programs and utilizing algorithms to accomplish a task. Students continue to decompose larger problems into smaller tasks and recognize the impacts of computing and computing devices. Students in fifth grade model how computing systems work. The accurate use of terminology as well as the responsible use of technology will continue to be built upon. The foundational understanding of computing and the use of technology will be an integral component of successful acquisition of skills across content areas.

## Algorithms and Programming

5.1 The student will construct algorithms (sets of step-by-step instructions) both independently and collaboratively,

a) using sequencing;
b) using loops; [Related SOL: Math 5.18]
c) using variables to store and process data; [Related SOL: Math 5.19]
d) performing number calculations on variables (addition, subtraction, multiplication and division,); and [Related SOL: Math 5.5, 5.7]
c) using conditionals (if-statements). [Related SOL: M 5.2, 5.3]

5.2 The student will construct programs to accomplish a task as a means of creative expression using a block or text based programming language, both independently and collaboratively

a) using sequencing;
b) using loops;
c) using variables;
e) using mathematical operations (addition, subtraction, multiplication and division) variable to manipulate a variable; and [Related SOL: Math 5.19]
f) using conditionals (if-statements).

5.3 The student will analyze and debug (correct and improve) an algorithm that includes sequencing, events, loops, conditionals, and variables. [Related SOL areas - Math: Problem Solving, English: Editing]

5.4 The student will create a plan as part of the iterative design process, both independently and collaboratively using strategies such as pair programming (e.g., storyboard, flowchart, pseudo-code, story map). [Related SOL: English 5.7 c, d, e]

5.5 The student will decompose (break down) a larger problem into smaller sub-problems, both independently and collaboratively. [Related SOL: Math 5.4]

5.6 The student will give credit to sources when borrowing or changing ideas (e.g., using information, pictures created by others, using music created by others, remixing programming projects). [Related SOL: English 5.9d, e]

## Computing Systems

5.7 The student will model how a computing system works including input and output, processors, sensors and storage.

5.8 The student will identify, using accurate terminology, simple hardware and software problems that may occur during use, and apply strategies for solving problems (e.g., rebooting the device, checking for power, checking network availability, closing and reopening an app).

## Cybersecurity

5.9 The student will evaluate and solve problems that relate to inappropriate use of computing devices and networks.

5.10 The student will determine whether passwords are strong, explain why strong passwords should be used, and demonstrate proper use and protection of personal passwords.

## Data and Analysis

5.11 The student will answer a question by using a computer to manipulate data in order for the student to draw conclusions and make predictions. [Related SOL: Math 5.16, 5.17, VS.1c and j]

5.12 The student will create an artifact using computing systems to model the attributes and behaviors associated with a concept (e.g., rocks). [Related SOL area - Math Models, VS.1c and j]

5.13 The student will use numeric values to represent non-numeric ideas in the computer (e.g., binary, ASCII, pixel attributes such as RGB). [Related SOL: Math 5.19a]

### Impacts of Computing

5.14 The student will give examples and explain how computer science had changed the world and express how computing technologies influence, and are influenced by, cultural practices.

5.15 The student will evaluate and describe the positive and negative impacts of the pervasiveness of computers and computing in daily life (e.g., downloading videos and audio files, electronic appliances, wireless Internet, mobile computing devices, GPS systems, wearable computing).

5.16 The student will explain social and ethical issues that relate to computing devices and networks. [Related SOL: C/T: 6-8.3]

### Networking and the Internet

5.17 The student will compare and contrast the difference between a local network and a worldwide Network.

# Grade Six

The sixth-grade standards emphasize constructing programs and utilizing algorithms to accomplish a task. Students continue to decompose larger problems into smaller tasks and recognize the impacts of computing and computing devices. Students in sixth grade begin to understand the means of storing data as representations of real world phenomena. The accurate use of terminology as well as the responsible use of technology will continue to be built upon. The foundational understanding of computing and the use of technology will be an integral component of successful acquisition of skills across content areas.

## Algorithms and Programming

6.1 The student will construct programs to accomplish a task as a means of creative expression or scientific exploration using a block based or text based programming language, both independently and collaboratively,

a) combining control structures such as if-statements and loops; and
b) creating clearly named variables that represent different data types, including numeric and non-numeric data, and perform operations on their values. [Related SOL: Math 6.3, 6.6]

6.2 The student will trace programs to predict outcomes and debug (correct and improve) for correctness.

6.3 The student will seek and incorporate feedback from team members and users to refine a program that meets user needs.

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

## Computing Systems

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

## Cybersecurity

6.6 The student will identify physical and digital security measures used protect electronic information.

## Data and Analysis

6.7 The student will explain how binary sequences are used to represent digital data. *Exclusion: Conversions between binary and base-ten numbers are beyond the scope of these standards.*

6.8 The student will collect data using computational tools then clean and organize to make it more useful and reliable.

6.9 The student will explain the insight and knowledge gained from digitally processed data by using appropriate visualizations.

6.10 The student will use models and simulations to formulate, refine, and test hypotheses.

## Impacts of Computing

6.11 The student will explain how computing has impacted innovations in other fields.

6.12 The student will explore careers relate to data. [Related SOL: English 6.6]

6.13 The student will explain why the speed of data transmission across the Internet can vary depending on the type of data being transmitted.

# Grade Seven

The seventh-grade standards emphasize constructing programs and utilizing algorithms to accomplish a task. Students continue to decompose larger problems into smaller tasks and recognize the impacts of computing and computing devices. Students in seventh grade explore processing data as well as its transmission over networks. The accurate use of terminology as well as the responsible use of technology will continue to be built upon. The foundational understanding of computing and the use of technology will be an integral component of successful acquisition of skills across content areas.

## Algorithms and Programming

7.1 The student will construct programs to accomplish a task as a means of creative expression or scientific exploration using a block based or text based programming language, both independently and collaboratively,

a) combining control structures such as if-statements and loops including compound conditionals; and
b) creating clearly named variables that represent different data types, including numeric and non-numeric data, and perform operations on their values. [Related SOL: Math 7.1, 7.2]

7.2 The student will document programs to make them easier to follow, test, and debug.

7.3 The student will distribute tasks and maintain a project timeline when collaboratively developing computational artifacts.

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

## Computing Systems

7.5 The student will describe how the Internet connects devices and networks all over the world. *Exclusion: Specific devices used to implement the Internet are beyond the scope of these standards.*

## Cybersecurity

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

7.7 The student will identify existing cybersecurity concerns and potential options to address these issues with the Internet and the systems that use it.

## Data and Analysis

7.8 The student will discuss the correctness of a model representing a system by comparing the model's generated results with data that were observed in the system being modeled.

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

## Impacts of Computing

7.10 The student will explain how advances in technology have contributed to Virginia's prosperity and role in the global economy. [Related SOL: VS.10]

7.11 The student will describe the development of new technologies in communication, entertainment, and business and their impact on American life. [Related SOL: USII.9]

7.12 The student will explore careers related to the Internet. [Related SOL: English 7.6]

## Networking and the Internet

7.13 The student will outline the advantages and disadvantages of transmitting information over the Internet, including speed, reliability, cost and security.

7.14 The student will explain why protocols are necessary in data transmission. Model the role of protocols in transmitting data across networks and the Internet.

7.15 The student will model how information is broken down into smaller pieces, transmitted as packets through multiple devices over networks and the Internet, and reassembled at the destination.

# Grade Eight

The eighth-grade standards emphasize constructing programs and utilizing algorithms to accomplish a task. Students continue to decompose larger problems into smaller tasks and recognize the impacts of computing and computing devices. Students in eighth grade continue to work with data including how it can be vulnerable and how it can be protected. The accurate use of terminology as well as the responsible use of technology will continue to be built upon. The foundational understanding of computing and the use of technology will be an integral component of successful acquisition of skills across content areas.

## Algorithms and Programming

8.1 The student will construct programs to accomplish a task as a means of creative expression or scientific exploration using a block based or text based programming language, both independently and collaboratively,

a) combining control structures such as if-statements and loops including nested conditionals and loops;
b) Using clearly named variables that represent different data types, including numeric and non-numeric data, and perform operations on their values; and [Related SOL: Math 7.1, 7.2]
c) Create procedures with parameters.

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

8.3 The student will explain how effective communication between participants is required for successful collaboration when developing programs.

8.4 The student will use flowcharts and/or pseudo code to address complex problems as algorithms.

## Computing Systems

8. 5 The student will, using the elements of computing devices such as primary memory, secondary storage, processor, input and output devices, and network connectivity; analyze the advantages and limitations of a given computing system.

## Cybersecurity

8.6 The student will identify physical and digital security measures used protect electronic information.

8.7 The student will identify impacts of hacking, ransomware, scams, fake vulnerability scans, and the ethical and legal concerns involved. *Exclusion: Students do not need to implement solutions.*

## Data and Analysis

8.8 The student will

a) explain the difference between a model and a simulation, and
b) create computational models to conduct simulations.

## Impacts of Computing

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

8.10 The student will evaluate online and print sources for appropriateness and credibility.

8.11 The student will discuss the social impacts and ethical considerations associated with the field of Cybersecurity.

8.12 The student will explore careers related to the field of Cybersecurity. [Related SOL: English 8.6]

**Networking and the Internet**

8.13 The student will identify existing cybersecurity concerns and potential options to address these issues with the Internet and the systems built on it.

**Middle School Computer Science Elective (MSCE) 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.

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

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

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

**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.

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.

**Additional Content for 18-week Module**
**Algorithms and Programming**

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

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

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

**Networks and the Internet**

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

**Cybersecurity**

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

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

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

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

**Impacts of Computing**

MSCSE.16 The student will discuss issues of bias and accessibility in the design of existing technologies.

MSCSE.17 The student will compare tradeoffs associated with computing technologies that affect people's everyday activities and career options.

MSCSE.18 The student will collaborate with many contributors through strategies such as crowdsourcing or surveys when creating a computational artifact or visualization.

MSCSE.19 The student will describe tradeoffs between allowing information to be public and keeping information private and secure.

**Computing Systems**

MSCSE.20 The student will systematically identify and correct problems with computing devices and their components.

MSCSE.21 The student will explore the relationship between hardware and software using the Internet of Things.

**36-week Module**

**Algorithms and Programming**

MSCSE.22 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.

MSCSE.23 The student will decompose problems and subproblems into parts to facilitate the design, implementation, and review of programs.

MSCSE.24 The student will create procedures with parameters to organize code and make it easier to reuse.

**Computing Systems**

MSCSE.25 The student will recommend improvements to the design of computing devices, based on an analysis of how users interact with the devices.

MSCSE.26 The student will design projects that combine hardware and software components to collect and exchange data.

# Computer Science Foundations

The Computer Science Foundations standards outline the content for a one-year course with an emphasis on computer programming within the context of broader concepts of computer science. The standards build on the concepts of computer science developed in prior grade levels. The standards provide a transition from block-based programming to a text-based programming language and familiarize the student with developing and executing computer programs. 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.

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.

## Computing Systems

CSF.1 The student will

a) compare the structures, functions, and interactions between application software, system software, and hardware; and
b) explore the relationship between hardware and software using the Internet of Things.

## Networks and the Internet

CSF.2 The student will model how information is broken down into smaller pieces, transmitted as packets through multiple devices over networks and the Internet, and reassembled at the destination.

CSF.3 The student will explain the role of protocols in transmitting data across networks and the Internet.

CSF.4 The student will evaluate the scalability and reliability of networks, by describing the relationship between routers, switches, servers, topology and addressing.

## Cybersecurity

CSF.5 The student will identify and explain ways that sensitive data (assets) can be threatened by malware and other computer attacks, using appropriate terminology.

CSF.6 The student will give examples of ways to protect sensitive data (assets) from malware and other computer attacks, and evaluate them according to multiple criteria.

CSF.7 The student will explain typical tradeoffs between usability and security, and recommend security measures in a given scenario based on these (or other) tradeoffs.

CSF.8 The student will write or adapt a program to validate its input, to avoid certain kinds of vulnerabilities.

## Data and Analysis

CSF.9 The student will evaluate the tradeoffs in how data elements are organized and where data is stored.

CSF.10 The student will create interactive data visualizations using software tools to help others better understand real-world phenomena.

CSF.11 The student will use data analysis tools and techniques to identify patterns in data representing complex systems.

## Algorithms and Programming

CSF.12 The student will develop a program working individually and in teams using a text-based language.

CSF.13 The student will identify the expected output of a program given a problem and some input.

CSF.14 The student will design and iteratively develop programs for practical intent or personal expression, incorporating feedback from users.

CSF15 The student will design and implement algorithms using
a) sequencing of instructions;
b) conditional execution; and
c) iteration.

CSF.16 The student will implement a program that accepts input values, stores them in appropriately named variables, and produces output.

CSF.17 The student will trace the execution of an algorithm, illustrating output and changes in values of named variables.

CSF.18 The student will apply the basic operations used with numeric and non-numeric data types in developing programs.

CSF.19 The student will use predefined functions to simplify the solution of a complex problem.

CSF.20 The student will apply simple algorithms to a collection of data.

CSF.21 The student will create programs
a) demonstrating an understanding that program development is an ongoing process that requires adjusting and debugging along the way; and
b) using version control to create and refine programs.

## Impacts of Computing

CSF.22 The student will use tools and methods for collaboration on a project to increase connectivity of people in different cultures and career fields.

CSF.23 The student will evaluate the ways computing impacts personal, ethical, social, economic, and cultural practices.

CSF.24 The student will explain the beneficial and harmful effects that intellectual property laws can have on innovation, including the impact of open source software.

CSF.25 The student will explain the privacy concerns related to the collection and generation of data through automated processes that are not always evident to users.

# Computer Science Principles

The Computer Science Principles standards outline the content for a one-year course with an emphasis on the principles underlying computer science. The standards build on the concepts outlined in the Computer Science Foundations standards.

Students in this course will expand their programming skills and begin to think about and analyze their own problem solving process. Students continue to develop the ideas and practices of computational thinking and consider how computing impacts the world.

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.

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.

## Computing Systems

CSP.1 The student will develop and apply criteria for evaluating a computer system for a given purpose.

CSP.2 The student will illustrate ways computing systems implement logic, input, and output through hardware components.

## Networks and the Internet

CSP.3 The student will explain abstractions enabling

a) one computer to communicate with another over an Internet connection; and
b) different layers of Internet technology to build on one another.

CSP.4 The student will explain design principles enabling large-scale operation of the Internet to connect devices and networks all over the world.

## Cybersecurity

CSP.5 The student will explain symmetric and asymmetric encryption as they pertain to messages being sent on a network.

## Data and Analysis

CSP.6 The student will discuss the methods and tradeoffs of collecting and analyzing data elements on a large scale.

CSP.7 The student will select data collection tools and techniques to generate data sets that support a claim or communicate information. Implement a relational database to work with data.

CSP.8 The student will discuss how data representations can be interpreted in a variety of forms, convert between data representations, and analyze the representation tradeoffs among various forms of digital information.

## Algorithms and Programming

CSP.9 The student will design and implement algorithms with

a) compound conditional execution; and
b) a variety of loop control structures.

CSP.10 The student will solve a complex problem by decomposing it into subtasks consisting of predefined functions and user-defined functions.

CSP.11 The student will store, process, and manipulate data contained in a data structure.

CSP.12 The student will systematically debug a program using an appropriate set of data.

## Impacts of Computing

CSP.13 The student will explain how computing has impacted innovations in other fields positively and negatively, and enables collaboration between a variety of people.

CSP.14 The student will evaluate the impact of equity, access, and influence on the distribution of computing resources in a global society, including the impacts of cloud computing.

CSP.15 The student will explain how intellectual property concerns affect the tools for and products of computing, including combining existing content to create new artifacts and the impact of open source and free software.

CSP.16 The student will evaluate the social and economic implications of privacy in the context of safety, law or ethics.

# Computer Science Programming

The Computer Science Programming standards outline the content for a one-year course with an emphasis on computer programming in a text-based language. The standards build on the concepts outlined in the Computer Science Foundations and Computer Science Principles standards.

This course continues the study of computer programming and prepares students to write programs of increasing complexity to solve problems of personal interest and professional relevance in a variety of technical fields. Additionally, this course provides the knowledge and experience to prepare students for further studies in computer science.

Teachers are encouraged to select text-based programming languages and environments, problems, challenges, and activities that are appropriate for their students to successfully meet the objectives of the standards. The majority of this course will address Algorithms and Programming. While the standards below do not include new content related to Computing Systems or Networks and the Internet, they may be used to provide context for additional exploration of these topics.

## Cybersecurity

PRG.1 The student will describe and use best practices of program development that make some common flaws less likely, and explain how this improves computer security.

## Data and Analysis

PRG.2 The student will create programs that model the relationships among different elements in collections of real-world data.

PRG.3 The student will translate numbers between machine representations and more human-accessible representations.

## Algorithms and Programming

PRG.4 The student will design and implement a program working individually and in teams using a text-based language.

PRG.5 The student will explain the software life cycle and how it applies to iterative development processes.

PRG.6 The student will design and implement an algorithm
- a) with compound conditional execution, and analyze and evaluate complex Boolean conditions; and
- b) using complex iteration, including nested loops.

PRG.7 The student will implement programs that accept input from a variety of sources and produce output based on that input.

PRG.8 The student will trace the execution of iterative and recursive algorithms, illustrating output and changes in values of named variables.

PRG.9 The student will perform complex computations
- a) on numbers, including modular division and random number generation; and
- b) on strings, including substring manipulation and processing individual characters.

PRG.10 The student will demonstrate an understanding of different data types by using appropriate constructs to convert between them when appropriate.

PRG.11 The student will analyze a large-scale computational problem, identify generalizable patterns, and implement a solution.

PRG.12 The student will implement an algorithm that uses existing functions and accesses existing libraries or APIs to satisfy its requirements.

PRG.13 The student will write functions, both with and without parameters, and both with and without return values, that represent abstractions useful to the solution of a larger problem.

PRG.14 The student will create programs demonstrating an understanding of the interactions between classes in object-oriented design, and by implementing classes with instance data and methods to satisfy a design specification.

PRG.15 The student will use code written by others by reading the documentation and incorporating it into their programs using proper citation of the reused code.

PRG.16 The student will read and store data in 1D and 2D collections, and design and implement algorithms to process and manipulate those collections.

PRG.17 The student will adapt classic algorithms for use in a particular context and analyze them for effectiveness and efficiency.

PRG.18 The student will develop and use a series of test cases to verify that a program performs according to its design specifications, including edge cases and all branches.

PRG.19 The student will, through the process of code review, evaluate a program's correctness, readability, usability, and other factors.

PRG.20 The student will use a systematic approach and debugging tools to independently debug a program.

**Impacts of Computing**

PRG.21 The student will identify some of the practical, business, and ethical impacts of open source and free software and the widespread access they provide.

Attachment B

# Virginia State Board of Education

## First Review of the Proposed 2017 *Computer Science Standards of Learning*

## July 27, 2017

Virginia Department of Education

# Terminology

## Computer Science

- Computer science is the study of computers and algorithmic processes, including their principles, their hardware and software designs, their applications, and their impact on society.

## Computational Thinking

- Computational Thinking - An approach to solving problems in a way that can be implemented with a computer. It involves the use of concepts, such as abstraction, recursion, and iteration, to process and analyze data, and to create real and virtual artifacts.

Source: Computer Science Teachers Association (CSTA) & Association for Computing Machinery (ACM)

# Terminology

## Educational Technology

- The use of computers across the curriculum, or more specifically, the use of computer technology (hardware and software) to learn about other disciplines.

## Information Technology

- The use of technologies to manipulate and share information in various forms. Information Technology involves learning about computers, but emphasizes the technology.

Source: Computer Science Teachers Association (CSTA)

# Strategic Plan for Computer Science Education Implementation

**5-Step Plan**

1. **Computer Science (CS) standards development**

2. **CS teacher training**
   a) **Curriculum, instruction, and assessment**
   b) **Licensure, endorsements, credentials**

3. **Increased course offerings**

4. **Consider CS career pathways**
   a) **Ensure career pathways**
   b) **Credentials choices**
   c) **Resources**

5. **Communications to public about CS and computing careers**

**SREB: Bridging the Computer Science Education Gap: Five Actions States Can Take**

VIRGINIA DEPARTMENT OF EDUCATION

# Computer Science Clarifications

1. Computer Science is only about getting a coding job.
2. Computer science is the same as digital literacy.
3. Computer Science is coding.
4. Computer Science is a high school subject.
5. Let’s just spread what we have to all.

# The Standards Development Process

- Information provided to the Steering committee
  - K-12 Computer Science Framework
  - Computer Science Teachers Association standards; revised in 2016
  - Existing state CS standards (AR, IN, MA, and WA)
  - Profile of a Virginia Graduate
  - Existing standards, including Computer Technology

# Inputs into the K-12 CS framework

- CS education research
- International standards (e.g., UK programme of study)
- CSTA 2011 standards; revised in 2016
- ACM CS 2013 (higher ed body of knowledge)
- Advanced Placement (AP) CS Principles course framework
- Math and Science frameworks and standards

# The Committees

- Steering committee: Grade band teams developed
  - (K-2, 3-5, 6-8, 9-12)
- Teacher Review committee: additional classroom teachers
- External Review committee:
  - convened two separate meetings to ensure business and industry feedback
  - 15 educational stakeholder organizations
  - 15 business and industry organizations

# The Computer Science Standards

- Grade level standards instead of grade band standards
  - K-5: Integrated standards
  - 6-8: Integrated and elective course standards
  - 9-12: Elective course standards

- 6 strands and 7 practices based upon K-12 CS Framework

# The Standards Development Process

Guidance provided to the Steering committee:

- Develop standards that …
  - show the vertical progression of computer science content and skills;
  - correlate to existing content area standards, including computer technology;
  - provide local school division flexibility in implementation;
  - ensure developmental appropriateness of student expectations; and
  - sufficiently prepare students for college and careers.

# The Computer Science Standards

- Correlate (horizontal alignment) to existing content area standards, including computer technology
  - Grade band teams reviewed existing standards in Mathematics, Science, History and Social Science, English, Family Life, and Computer Technology
  - High level application of content and skills desired along with appropriate rigor; balance of the two
  - Allow for integration of CS standards into existing content areas as done with Computer Technology standards

# The Computer Science Standards

- Vertical progression of computer science content and skills
  - Grade band teams developed standards; exchanged with other teams during Steering committee and Teacher Review committee meetings
  - High level application of content and skills desired along with appropriate rigor; balance of the two
  - Higher ed organizations along with business and industry see potential for standards to advance the level of instruction and training provided after high school to students and workers.

# The Computer Science Standards

- Provide local school division flexibility in implementation
  - Integrated and elective standards developed for middle school implementation (provides guidance with flexibility)
  - Middle School Computer Science Elective (MSCE) standards for modules (not courses)
  - Multiple options for implementation (6, 9,18, 36 weeks)
  - Core 6-week module with additional content added for additional weeks of implementation
  - HS course standards developed for three levels
    - Provides a more gradual transition to AP level CS courses

# The Computer Science Standards

- Ensure developmental appropriateness of student expectations
  - High level application of content and skills desired along with appropriate rigor; balance of the two
  - Verb usage consistent with existing standards, but also more focused on application of knowledge and skills and student-centered learning

# The Computer Science Standards

- Sufficiently prepare students for college and careers
  - The outcomes of these standards will require several years before the elementary and middle school impacts are realized

# Questions and Comments?