Digital Readiness

K-8 Computer Science Standards

Tennessee Department of Education | March 2018

Introduction

Tennessee students have various 21st century needs that their K-12 education should address in order to enable them to engage and thrive in a connected, digital world. The Tennessee K-8 Computer Science Standards are designed to be used within and across all content areas in order to enhance learning and to ultimately inspire students to take ownership of their own academic success.

> *The Tennessee K-8 Computer Science Standards lay a foundation in a continuously evolving technological world.*

Computer science is one of the **fastest growing industries** and computer programmers are needed within every field, including healthcare, transportation, and banking.[1] Although not every Tennessee student will enter a STEM field, all students will benefit from learning computer science concepts and practices allowing them to better understanding the world around them, improve their logical reasoning and problem-solving skills, and increase their creativity and collaboration.

ACT results over the past two years in Tennessee indicate **that close to half (48%) of the state's graduates have an interest in pursuing a STEM field**, but **only 21% meet the ACT STEM benchmarks**. While the number of students meeting the STEM benchmark has improved by a few percentage points since 2013, Tennessee continues to see gaps between interest and aptitude for STEM. Additionally, STEM interest has declined since 2013, with minority and female interest particularly low. One way to shrink this gap is to explicitly focus on how STEM content is applied in the classroom. Because computer science is an applied science it encourages students to understand and engage in the "why" of science, technology, engineering and math, so they can begin to see connections to careers. This can serve as a needed inspiration as students learn science and math.

In addition to the benefits for students, investing in computer science will directly benefit Tennessee's economy. Tennessee, like many other states, has seen a steady increase in the number of jobs in the computer science and technology fields. From 2013 to 2015, Tennessee's **advanced industry jobs increased by an average of 4.6 percent** annually, outpacing the national average, and the Nashville metropolitan region **ranked No. 1**

[1] K12 Computer Science

among the 100 largest metro areas in the U.S. **for advanced industry job growth**.[2] Specifically, IT employment throughout the state grew 13.7% from 2010-2015.

In order for Tennessee to continue its economic growth, it must take advantage of the growing number of high wage technology jobs available by ensuring that a skilled workforce is available and ready to meet the demand. Currently, Tennessee **employment in the IT field is 39% below the national average**, and during 2015, there were **89,686 unique job postings for IT jobs in the state but only 32,052 hires**. Employers cite **IT occupations as one of the four most in-demand occupations** across the state, and we must do more to meet this demand.[3]

The instructional focus in elementary school will be introducing students to computer systems and helping them develop an awareness of safe and responsible device practices, such as protecting private information and best practices for creating and using passwords. Students will also develop an understanding of how multi-step solutions are executed within computer programs. The instructional focus in middle school will extend digital citizenship discussions to include safely interacting with people and content online. Students will engage in activities to introduce computational thinking and information-processing skills to be utilized in creating programs. Students will explore connections between computer science skills and career opportunities. At the high-school level, students have the option of taking either elective coursework (i.e., Computer Science Foundations or AP Computer Science Principles) or enrolling in a technology-focused career and technical education program of study which includes the opportunity for students to gain multiple industry-recognized certifications. These high school courses are focused on developing industry specific skills through the application of technical skills to real world problems.

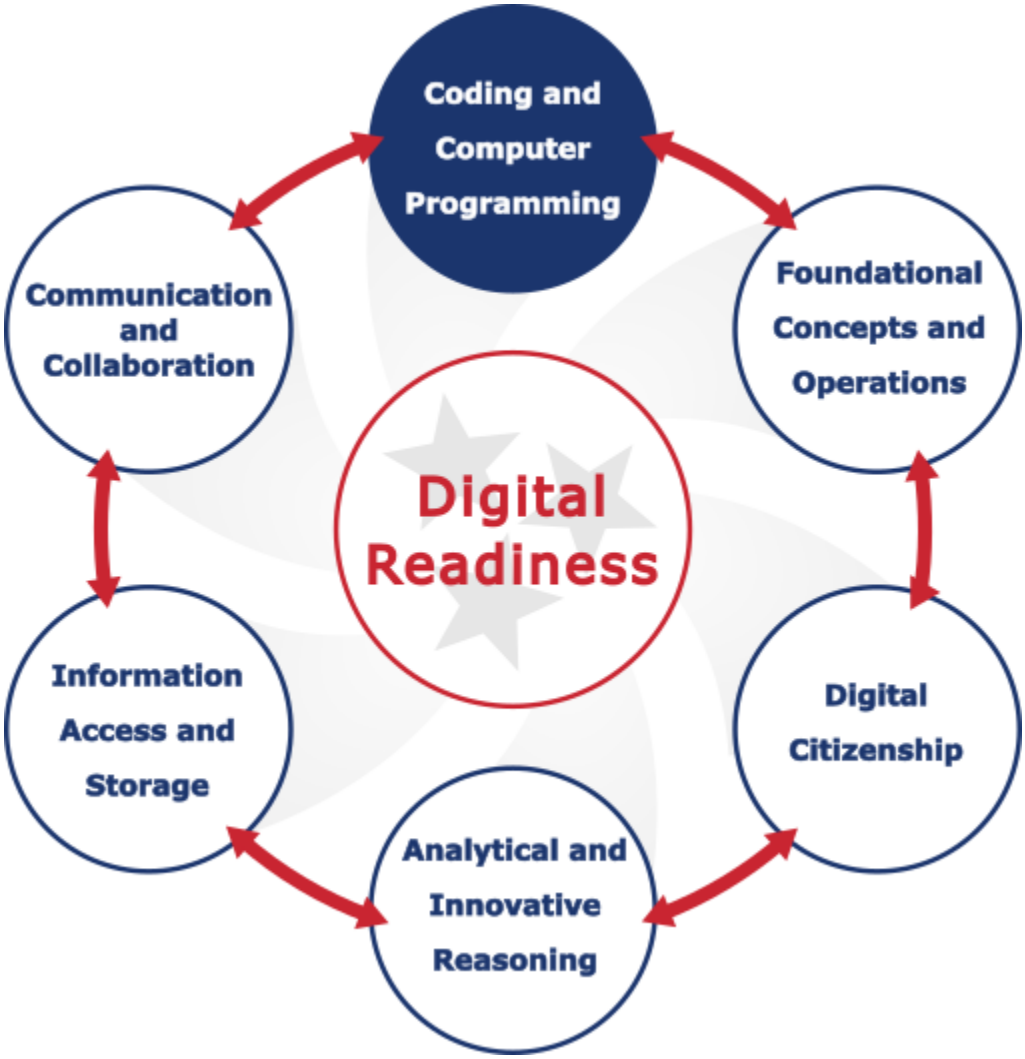
The standards are divided into six major strands, with each strand developing from one grade level to the next and extending into a student's high-school course work. The five

[2] https://www.tn.gov/ecd/news/44561

[3] TN ECD LEAP 2016 Report

digital readiness strands are not meant to be taught in isolation, rather they should be integrated within grade-level content areas with a goal of cultivating empowered learners throughout their academic career. The Coding and Computer Programming strand should be addressed through explicit computer science instruction. The Tennessee K-8 Computer Science Standards lay a foundation that enables students to be workforce and post-secondary ready in a continuously evolving technological world.

The six strands for the K-8 Computer Science standards include:



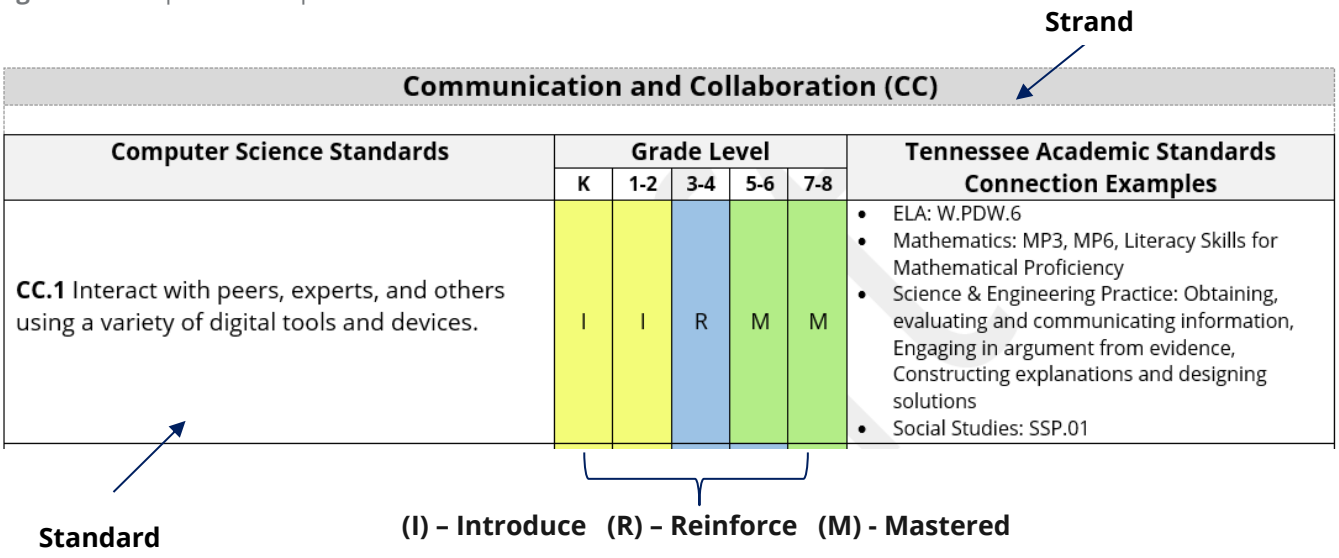
Strands

- **Foundational Concepts and Operations -** Demonstrate proficiency in the use of computers and applications as well as an understanding of the concepts underlying hardware, software, and connectivity.
- **Analytical and Innovative Thinking -** Use critical thinking skills to plan and conduct research, manage projects, solve problems, and make informed decisions using appropriate digital tools and resources.
- **Information Storage and Access -** Apply digital tools to store, access, and use information in a variety of capacities to enhance learning.
- **Communication and Collaboration –** Use digital media and environments to communicate and work collaboratively to support individual learning and contribute to the learning of others.
- **Digital Citizenship -** Demonstrate the appropriate use of technology and an understanding of ethical behavior and safety issues in an interconnected digital society.
- **Coding and Computer Programming –** Use analytical and innovative problem-solving skills to decompose, identify patterns, generalize information, and formulate algorithmic processes to solve a problem or related set of problems with a variety of tools.

The five digital readiness strands consist of **Foundational Concepts and Operations** (FCO), **Analytical and Innovative Thinking** (AIT), **Information Storage and Access** (ISA), **Communication and Collaboration** (CC), and **Digital Citizenship** (DC), which are divided into K-8 standards, guidelines for embedded instruction, and connections to the core content standards. Each of these standards span across all grade levels and is denoted as introduced, reinforced, or mastered, based on age appropriateness and content coherence.

Example from the K-8 Computer Science Standards Document

Figure 1. Example K-8 Computer Science Standards document.



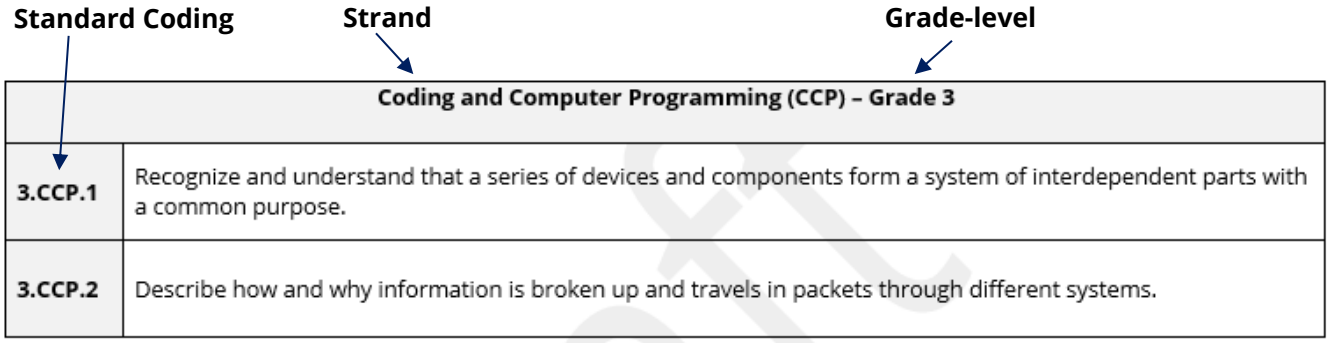
The digital readiness strand is indicated at the top of the table of standards. The left column of the table contains the topic heading and the standard is listed in the middle column. The standards document indicates the grade level when each skill is generally Introduced (I), Reinforced (R), and Mastered (M). The right column indicates connections between the Computer Science standard and the content area standards for ELA, math, science, and social studies.

The connections provided in the example above indicate direct connections to technology and digital tools that are aligned to the content standards. Some academic standards provide direct correlations to technology, whereas other standards indirectly lend themselves to embedding technology within the content. The connections listed in the standards document are examples aligned to the content standards, but do not provide an exhaustive list. Both connections are meant to naturally blend technology and content to enhance learning for students. The International Society of Technology Education states, "Ultimately, it's not about the technology at all. It's about changing the way learning and teaching takes place to make it more meaningful and impactful for educators and learners around the globe. It's about working together to turn what-ifs into what is."

> *"It's about changing the way learning and teaching takes place to make it more meaningful and impactful for educators and learners around the globe."*
>
> *ISTE, 2016*

The **Coding and Computer Programming** (CCP) strand is sub-divided into two sets of standards. Grades K-5 consist of a set of standards for each grade level. Grades 6-8 consist of a cummlative set of standards to be taught over the course of the middle grades in a manner to be determined by each district. The Coding and Computer Programming standards are designed as separate content to support students in breaking apart problems, observing the interconnectness of their parts, and using knowledge of patterns to design solutions with a variety of tools including coding. This coding and programming strand is developed to complement existing math and science standards.

Example from the Coding and Computer Programming Standards



Coding and Computer Programming (CCP) – Grade 3	
3.CCP.1	Recognize and understand that a series of devices and components form a system of interdependent parts with a common purpose.
3.CCP.2	Describe how and why information is broken up and travels in packets through different systems.

Figure 2. Coding and Computer Programming example from the Digital Readiness Computer Science Standards document.

Each **Coding and Computer Programming** (CCP) table includes the strand and grade-level in the top row. The left column of the table indicates the standard coding and the standard is located in the right column of the table. These standards are divided into grade-level specific standards for grades K-5 and a set of standards for grades 6-8.

Example coding for the Coding and Computer Programming strand:
3.CCP.1
3 is the grade level
CCP is the coding abbreviation for Coding and Computer Programming
1 is the standard number in consecutive numerical order

"As educators, we are preparing students for a future that we cannot yet imagine. Empowering students to become lifelong learners and providing them with the skills to face future challenges resourcefully and creatively is critical. It's not about using digital tools to support outdated education strategies and models; it's about tapping into technology's potential to amplify human capacity for collaboration, creativity and communication. It's about leveling the playing field and providing young people worldwide with equitable access to powerful learning opportunities." (ISTE, 2016)

Digital Readiness

K-8 Computer Science Standards

Foundational Concepts and Operations (FCO)

Computer Science Standards	Grade Level					Tennessee Academic Standards Connections Examples
	K	1-2	3-4	5-6	7-8	
FCO.1 Demonstrate fundamental technology skills (e.g., turn on and login to device).	I	I	R	M	M	• Mathematics: MP1, MP5 • Science Crosscutting Concepts**:** Pattern; Cause and effect
FCO.2 Interact with a device using a pointing tool such as a mouse, tactile sensor, or other input.	I	R	R	M	M	• Mathematics**:** MP1, MP5 • Science Crosscutting Concept: Structure and function
FCO.3 Navigate to applications and documents by using desktop icons, windows, and menus.	I	I	R	M	M	• Mathematics**:** MP1, MP5 • Science & Engineering Practice: Using mathematics and computational thinking
FCO.4 Use age-appropriate online tools and resources (e.g., tutorial, assessment, web browser).	I	I	R	M	M	• Mathematics**:** MP1, MP5 • Science & Engineering Practice: Obtaining, evaluating and communicating information
FCO.5 Demonstrate fundamental keyboarding skills.	I	I	R	R	M	• ELA: FL.PC.1 • Mathematics**:** MP1, MP5
FCO.6 Select and use appropriate word processing, spreadsheets, and multimedia applications.	I	I	R	R	M	• Mathematics**:** MP1, MP4, MP5, MP8 • Science & Engineering Practice: Using mathematics and computational thinking
FCO.7 Use menu, tool bar, and editing functions (e.g., font/size/style/line spacing, margins, spell check) to format, edit, save, and print a document.	I	I	R	R	M	• ELA: FL.WC.4 • Mathematics: MP5, MP6
FCO.8 Identify and solve routine hardware and software problems that occur during routine usage.	I	I	R	M	M	• Mathematics: MP1, MP4 • Science & Engineering Practices: Planning and carrying out controlled investigations; Constructing explanations and designing solutions • Science Crosscutting Concepts: Pattern; Cause and effect

Communication and Collaboration (CC)

Computer Science Standards	Grade Level					Tennessee Academic Standards Connection Examples
	K	1-2	3-4	5-6	7-8	
CC.1 Interact with peers, experts, and others using a variety of digital tools and devices.	I	I	R	M	M	• ELA: W.PDW.6 • Mathematics: MP3, MP6, Literacy Skills for Mathematical Proficiency • Science & Engineering Practices: Obtaining, evaluating and communicating information; Engaging in argument from evidence; Constructing explanations and designing solutions • Social Studies: SSP.01
CC.2 Communicate information and ideas effectively to multiple audiences using a variety of media and formats. (e.g., reports, research papers, presentations, newsletters, Web sites, podcasts, blogs), citing sources.	I	I	R	R	M	• ELA: SL.PKI.4, SL.PKI.5, R.RI.IKI.7 • Mathematics: MP3, MP6, Literacy Skills for Mathematical Proficiency • Science & Engineering Practice: Obtaining, evaluating and communicating information • Social Studies: SSP.01, SSP.04
CC.3 Contribute, individually or as part of a team, to work to identify and solve authentic problems or produce original works using a variety of digital tools and devices.	I	I	R	R	M	• ELA: SL.CC.1 • Mathematics: MP2, MP3, MP4, MP5, MP6, MP7, Literacy Skills for Mathematical Proficiency • Science & Engineering Practices: Asking questions and defining problems; Developing and using models; Analyzing and interpreting data; Using mathematics and computational thinking; Constructing explanations and designing solutions

Analytical and Innovative Thinking (AIT)

Computer Science Standards	Grade Level					Tennessee Academic Standards Connection Examples
	K	1-2	3-4	5-6	7-8	
AIT.1 Identify and define problems and form significant questions for investigation.	I	R	R	M	M	• ELA: RL.KID.1 • Mathematics: MP1, MP2, MP4, MP7, MP8 • Science & Engineering Practice: Asking questions and defining problems • Social Studies: SSP.03
AIT.2 Develop a plan to use technology to find a solution and create projects.	I	I	R	R	M	• ELA: SL.CC.2, W.PDW.6 • Mathematics: MP1, MP2, MP8 • Science & Engineering Practice: Planning and carrying out controlled investigations, constructing explanations and designing solutions
AIT.3 Determine the best technology and appropriate tool to address a variety of tasks and problems.	I	I	R	R	M	• ELA: SL.CC.2, W.PDW.6 • Mathematics: MP5, MP6 • Science & Engineering Practice: Using mathematics and computational thinking
AIT.4 Use multiple processes and diverse perspectives to explore alternative solutions.	I	R	R	M	M	• ELA: SL.CC.2, SL.CC.3, R.RI.CS.6 • Mathematics: MP1,MP4,MP8 • Science & Engineering Practices: Using mathematics and computational thinking; Engaging in argument from evidence; Obtaining, evaluating, and communicating information • Social Studies: SSP.1, SSP.02, SSP.04
AIT.5 Evaluate the accuracy, relevance, appropriateness, and bias of electronic information sources.	I	I	I	R	M	• ELA: SL.CC.2, W.TTP.1, W.TTP.2, W.PDW.6, R.RI.IKI.8 • Mathematics: MP1,MP8 • Science & Engineering Practices: Engaging in argument from evidence;

						Obtaining, evaluating, and communicating information • Social Studies: SSP.02, SSP.03
AIT.6 Collect, organize, analyze, and interpret data to identify solutions and/or make informed decisions.	I	I	R	R	M	• ELA: SL.CC.2, W.TTP.1, W.TTP.2, W.PDW.6 • Mathematics: MP6, MP7, MP8 • Science & Engineering Practices: Analyzing and interpreting data; Constructing explanations and designing solutions • Social Studies: SSP.1, SSP.02, SSP.03, SSP.04
AIT.7 Infer and predict or propose relationships with data.	I	I	R	R	M	• ELA: SL.CC.2, R.RI.IKI.8 • Mathematics: MP1, MP6 • Science & Engineering Practices: Analyzing and interpreting data; Constructing explanations and designing solutions; Engaging in argument from evidence
AIT.8 Identify that various algorithms can achieve the same result and determine the most efficient sequence.	I	I	R	R	M	• Mathematics: MP1, MP2, MP4, MP7, MP8 • Science & Engineering Practice: Using mathematics and computational thinking

Digital Citizenship (DC)

Computer Science Standards	Grade Level					Tennessee Academic Standards Connection Examples
	K	1-2	3-4	5-6	7-8	
DC.1 Advocate, demonstrate and routinely practice safe, legal, and responsible use of information and technology.	I	I	R	M	M	• ELA: W.TTP.1 • Mathematics: MP5 • Science & Engineering Practice:, Obtaining, evaluating, and communicating information
DC.2 Exhibit a positive mindset toward using technology that supports collaboration, learning, and productivity.	I	R	R	M	M	• ELA: SL.CC.1, W.PDW.6 • Science & Engineering Practices: Engaging in argument from evidence; Obtaining, evaluating, and communicating information
DC.3 Exhibit leadership for digital citizenship.	I	R	R	M	M	
DC.4 Recognize and describe the potential risks and dangers associated with various forms of online communications (e.g., cell phones, social media, digital photos).	I	R	R	M	M	• ELA: R.KID.2, R.KID.3, R.RI.IKI.8, W.TTP.2 Mathematics: MP2 • Science & Engineering Practices: Engaging in argument from evidence; Obtaining, evaluating, and communicating information
DC.5 Explain responsible uses of technology and digital information; describe possible consequences of inappropriate use such as copyright infringement and piracy.	I	R	R	M	M	• ELA: R.KID.2, R.KID.3, R.RI.IKI.8, W.TTP.2, W.PDW.6 • Mathematics: MP3 • Science & Engineering Practice: Obtaining, evaluating, and communicating information

Information Storage and Access (ISA)

Computer Science Standards	Grade Level					Tennessee Academic Standards Connection Examples
	K	1-2	3-4	5-6	7-8	
ISA.1 Enter, organize, and synthesize information in a variety of platforms. (e.g., saving, organizing, and storing word documents and spreadsheets)	I	R	R	R	M	• ELA: R.CS.5, R.IKI.9 • Mathematics: MP5, 1.MD.C.5 • Science & Engineering Practices: Developing and using models; Analyzing and interpreting data; Constructing explanations and designing solutions; Obtaining, evaluating, and communicating information • Social Studies: SSP.3
ISA.2 Identify and use a variety of storage media and demonstrate an understanding of the rationale for using a certain medium for a specific purpose.	I	I	R	R	M	• ELA: R.CS.6 • Mathematics: MP5 • Science & Engineering Practice: Using mathematics and computational thinking
ISA.3 Plan and use strategies to access information and guide inquiry.	I	I	R	R	M	• ELA: RL.KID.1 • Mathematics: MP1 • Science & Engineering Practice: Obtaining, evaluating, and communicating information
ISA.4 Locate information from a variety of sources.	I	I	R	R	M	• ELA: R.KID.1, R.IKI.7 • Mathematics: MP5 • Science & Engineering Practice: Obtaining, evaluating, and communicating information • Social Studies: SSP.1
ISA.5 Perform basic searches on databases to locate information.	I	I	I	I	R	• ELA: R.KID.2, R.KID.3 • Mathematics: MP1 • Science & Engineering Practice: Obtaining, evaluating, and communicating information

ISA.6 Select appropriate information sources and digital tools.	I	R	R	M	M	• ELA: R.RI.IKI.8 • Mathematics: MP5 • Science & Engineering Practices: Engaging in Argument from Evidence; Obtaining, evaluating, and communicating information
ISA.7 Use age appropriate technologies to locate, collect, organize content from media collection(s) for specific purposes, such as citing sources.	I	R	R	M	M	• ELA: R.CS.5, R.RI.IKI.8 • Mathematics: MP5 • Science & Engineering Practice: Obtaining, evaluating, and communicating information • Social Studies: SSP.1, SSP.03
ISA.8 Describe the rationale for various security measures when using technology.	I	I	R	M	M	• ELA: R.KID.2, R.RI.IKI.8, W.TTP.2 • Mathematics: MP3

Coding and Computer Programming (CCP) – Grade K	
K.CCP.1	Identify, using appropriate terminology, common physical components of computing systems (hardware). *For example, but not limited to, desktop computers, laptop computers, tablet devices, monitors, keyboards, mice and printers.*
K.CCP.2	Use simple trial and error strategies to identify when a computing device is not working as intended. *For example, but not limited to, if the device does not turn on students can identify if it needs to be charged or is unplugged before saying the device does not work.*
K.CCP.3	Define an algorithm as a list of steps that can be followed to finish a task or solve a problem.
K.CCP.4	Decompose an example problem into smaller sub-problems with teacher guidance or independently.
K.CCP.5	Collaboratively, students can build independence and sophistication using a simple design process *(e.g., Ask, Plan, Do, Reflect)* to illustrate a program's sequence and outcomes.

Coding and Computer Programming (CCP) – Grade 1	
1.CCP.1	Recognize and utilize common physical components of computing systems (hardware) and software concepts using correct terminology. *For example, but not limited to, laptop computers, tablets, monitors, keyboards, printers and software concepts such as, sign-in requirements input output, debug and program.*
1.CCP.2	Use simple trial and error strategies to identify hardware and software problems that occur using appropriate terminology. *For example, but not limited to, an app or program is not working as expected, no sound is coming from a device, or a device will not turn on.*
1.CCP.3	Construct an algorithm by arranging sequential events step-by-step in a logical order.
1.CCP.4	Determine that data *(e.g., numbers, words, colors, and images)* can be stored in computer programs.
1.CCP.5	Collaboratively or individually, students use programming to create simple animated stories or solve pre-existing problems using a precise sequence of instructions and simple loops. *For example, but not limited to, if a dialogue is not sequenced correctly, the animated story will not make sense or using loops in a program to show the life cycle of a butterfly, a loop could be combined with move commands to allow continual but controlled movement of the character.*
1.CCP.6	Decompose larger problems or tasks into smaller sub-problems independently.
1.CCP.7	Collaboratively, students can build independence and sophistication using a simple design process *(e.g., Ask, Plan, Do, Reflect)* to illustrate a program's sequence and outcomes.
1.CCP.8	Compare positive and negative effects computer technology has in the lives of people. Identify ways that programs and/or hardware is used by groups within society. For example, touchscreens are used by children differently than they are used by artists.

Coding and Computer Programming (CCP) – Grade 2	
2.CCP.1	Identify and describe how hardware and software components make up a computing system.
2.CCP.2	Identify, using accurate terminology and debugging strategies, simple hardware and software problems that may occur during use. *For example, but not limited to, if an app or program is not working as expected, no sound or device won't turn on.*
2.CCP.3	Analyze and improve an algorithm that includes sequencing and simple patterns with or without a computing device.
2.CCP.4	Evaluate how computer programs can manipulate stored data *(words, numbers, colors, and images)* with support or independently.
2.CCP.5	Create or revise a computational artifact *(a visualization, a graphic, a video, a program, or an audio recording),* using appropriate attributions for revisions.
2.CCP.6	Define a problem or task, decompose it into smaller sub-problems.
2.CCP.7	Collaboratively, students can build independence and sophistication using a simple design process *(e.g., Ask, Plan, Do, Reflect)* to construct a program's sequence and revise outcomes.
2.CCP.8	Compare positive and negative impacts effects computer technology has in the lives of people.

Coding and Computer Programming (CCP) – Grade 3	
3.CCP.1	Recognize and understand that a series of devices and components form a system of interdependent parts with a common purpose.
3.CCP.2	Describe how and why information is broken up and travels in packets (collections of data).
3.CCP.3	Identify and determine the purpose of a variable and the data that it stores in an algorithm.
3.CCP.4	Using a block of code or script from a previous program, identify the control structures in the algorithm such as loops, and/or conditionals in the code.
3.CCP.5	Using a block of code or script from a previous program, decompose into sections and/or subprograms to make it easier to read or more manageable.
3.CCP.6	Using a block of code or script from a previous lesson, identify sections for the code that may be reused into a new strand of code.
3.CCP.7	Describe ways that programs and/or hardware are used by groups within society. *For example, touchscreens are used by children differently than they are used by artists.*

Coding and Computer Programming (CCP) – Grade 4	
4.CCP.1	Recognize the input and output devices along with the components that form an interdependent system with a common purpose.
4.CCP.2	Demonstrate how information is broken up and can travel in packets through different systems.
4.CCP.3	Using a block of code or script from an existing program, identify the variables in the algorithm to determine if or how these might be manipulated to improve the program.
4.CCP.4	Construct an algorithm to solve a problem that includes control structures such as loops, event handlers, and conditionals collaboratively with or without a computing device.
4.CCP.5	Using a block of code or script that has been used in a previous program or algorithm, identify sections that can be reused into a new block or script of code.
4.CCP.6	Use existing code and identify sections of code that can be used to remix into a new program with proper attributions for efficiency.
4.CCP.7	Describe ways that hardware and software are used by various members of society including accessibility features. For example, voice commands can be used for accessibility or convenience.

Coding and Computer Programming (CCP) – Grade 5	
5.CCP.1	Identify and describe the role of various input and output devices and components that are within an interdependent system with a common purpose.
5.CCP.2	Investigate and trace a bundle of information through a series of packets and different systems via a protocol.
5.CCP.3	Decompose (break down) complex real-world problems in multiple ways that use variables to develop a solution or procedure based on data.
5.CCP.4	Create an algorithm which includes control structures to solve a problem using visual block-based and/or text-based programming language both collaboratively and individually.
5.CCP.5	Decompose complex code into subsections or subprograms for reuse into other programs.
5.CCP.6	Decompose a piece of code with the intent to debug a section of code.
5.CCP.7	Formulate alternative uses for software and hardware for various members of society.

Coding and Computer Programming (CCP) – Grade 6-8	
CCP.1	Identify the advantages, disadvantages and unintended consequences of computing devices.
CCP.2	Analyze the relationship between human and computer interactions to improve the device. *For example, student A watches student B use a simple communication device. Student A updates the tool for improved use.*
CCP.3	Identify and describe multiple considerations and tradeoffs when designing or selecting computing system, such as functionality, cost, size, speed, accessibility, and aesthetics.
CCP.4	Construct optimized models of computing systems.
CCP.5	Create structured processes to troubleshoot problems with computing systems.
CCP.6	Define protocols in relation to a set of rules.
CCP.7	Construct protocols that can be used to share information between people or devices. *For example, a binary communication protocol using lights.*
CCP.8	Compare the relative strengths and weaknesses of unique protocols considering security, speed, and reliability.
CCP.9	Create models of networks that include packets and domain name server (DNS).

CCP.10	Identify steps to ensure security measures are in place to safeguard online information.
CCP.11	Create cyphers to encrypt data that can be transferred between users.
CCP.12	Explain how encryption can be used to safeguard data that is sent across a network.
CCP.13	Evaluate the accuracy and precision of various forms of data collection.
CCP.14	Identify and define the limiting factors to specific forms of data collection.
CCP.15	Describe how different formats of stored data represent tradeoffs between quality and size.
CCP.16	Represent data using different encoding schemes, such as binary, Unicode, Morse code, shorthand, student-created codes.
CCP.17	Explain the processes used to collect, transform, and analyze data to solve a problem using computational tools.
CCP.18	Revise variables and constants in computational models to more accurately reflect real-world systems. For example in an ecosystem model, introducing predators as a new variable.

CCP.19	Solicit and integrate peer feedback as appropriate to develop or refine a program.
CCP.20	Compare different algorithms that may be used to solve the same problem in terms of their speed, clarity, and size.
CCP.21	Provide proper attribution when code is borrowed or built upon.
CCP.22	Interpret the flow of execution of algorithms and predict their outcomes.
CCP.23	Design, develop, and present computational artifacts such as mobile applications that address social problems both independently and collaboratively.
CCP.24	Develop programs, both independently and collaboratively, that include sequences with nested loops and multiple branches. *(Clarification: At this level, students may use block- based and/or text-based programming languages.)*
CCP.25	Identify the purpose of variables in relation to programming
CCP.26	Create variables that represent different types of data and manipulate their values.
CCP.27	Define and use procedures that hide the complexity of a task and can be reused to solve similar tasks. *(Clarification: Students use and modify, but do not necessarily create, procedures with parameters.)*

CCP.28	Decompose a problem into parts and create solutions for each part.
CCP.29	Use an iterative design process *(e.g., define the problem, generate ideas, build, test, and improve solutions)* to solve problems, both independently and collaboratively.
CCP.30	Analyze the positive and negative impacts of computing technology.
CCP.31	Recognize there are tradeoffs in computing.
CCP.32	Explain how social interactions can allow for multiple viewpoints.
CCP.33	Demonstrate an understanding of digital security.