The Coding and Computer Programming strand should be addressed through explicit computer science instruction.

The Tennessee Computer Science Standards do a great job connecting the digital readiness strands to standards from other disciplines, with the exception of the Coding and Computer Programming strand. This page mentions three additional little ideas about particular CS standards.

TN CS Standard		
ISA.1 Enter, organize, and synthesize information in a variety of platforms. (e.g., saving, organizing, and storing word documents and spreadsheets)	Introduce: Students have a folder that travels from home to school. We enter information by placing documents in the folder. The folder organizes documents into "keep at home" and "return to school." After students are familiar with the folder, discuss the folder with students in these terms.	Introduce: Whether it is paper or electronic, and whether students or teachers do the data entry, we keep grade books. We enter and then retrieve the grades. We sythesisze the information into report cards.
ISA.4 Locate information from a variety of sources	Reinforce: It is critical that students understand that "a variety of sources" does not mean "more than one website." For a given topic, many websites will have copies of the same exact text (frequently from Wikipedia).	

This matrix describes some connections between Tenneessee Computer Science standards and Tennessee standards from other subjects. It is not an exhaustive list but shows examples of how computer science instruction and other topics can be incorporated into a single lesson.

Times New Roman font indicates text from TN Standards Documents

Calibri font indicates something added at the STEM Center

		Mathematics	Science	ELA	Social Science
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.CC.A.1 Count to 100 by ones, fives, and tens . . Count components; count keys on a keyboard; QWERTY keyboard has five keys per column (1,Q,A,Z,Alt) count by fives until spacebar K.CC.B.5 "How many keys in the top row?" "How many cords?	CCC: Scale If an older computer with separate monitor/speakers/keyboard is available, demonstrate how all of these parts are miniaturized in a tablet or smartphone. Also, K.LS1.2 computers use energy and can even "talk," but are they living?	K.FFL.SC.6.a With modeling or verbal prompts, orally produce complete sentences. "The keyboard is in front of the computer." "The mouse is blue."	K.06 Describe jobs in which workers frequently use computers vs. jobs where they do not
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.*		CCC: Cause and effect	K.FFL.SC.6.e Understand and use question words (interrogatives) when speaking and in shared language activities. "Where is the plug?" "When was the last time it worked?"	

K.CCP.3	Define an algorithm as a list of steps that can be followed to finish a task or solve a problem.		K.PS1.3 Construct an evidence-based account of how an object made of a small set of pieces (blocks, snap cubes) can be disassembled and made into a new object. Describe this process as an algorithm.		
K.CCP.4	Decompose an example problem into smaller sub-problems with teacher guidance or independently.	K.OA.A.3 or K.MD.C.4 Sort a pile of yellow and red rectangles and triangles first by separating red from yellow and then separating the yellow pile into shapes and the red pile into shapes. Notice that you can pass the yellow pile to a partner for shape-sorting			
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.				

1.CCP.1	Recognize and utilize common physical components of computing systems (hardware) and software concepts using correct terminology. *For example,. . .software concepts such as sign-in requirements, input-output, debug and program*	For 3 + 2 = ___ 3 and 2 are inputs and 5 is the output For 3 + __ = 5 3 and 5 are the inputs and 2 is the output			1.04 Give examples of products (goods) that people buy and use. Notice that software is a (good) even though we cannot usually see it.
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				
1.CCP.4	Determine that data *(e.g., numbers, words, colors, and images)* can be stored in computer programs.			3.FL.WC.4.C Look at an online dictionary	1.10 Recognize basic map symbols, including: cities, land, roads, and water. Look at such features in Mapquest or Google maps or a similar program
1.CCP.5	Collaboratively or individually, students use programming to create simple animated stories or solve preexisting 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.RL.KID.2 1.RL.KID.3 1.RL.CS.6 (character telling story moves)	1.21 Identify Tennessee Symbols Write the story about "This is the state flag." "This is the state tree." etc., and get the text or sound synchronized with pictures

1.CCP.6	Decompose larger problems or tasks into smaller sub-problems independently.	1.MD.C.5			
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.				1.25 Compare ways people lived in the past to how they live today, including: forms of communication, modes of transportation, and types of clothing.
2.CCP.1	Identify and describe how hardware and				
2.CCP.2	Identify, using accurate terminology and		2.ETS1.2 Sketch how		
2.CCP.3	Analyze and improve an algorithm that includes sequencing and simple patterns with or without a computing device.		2.LS3.1 Use a computer simulation to determine numbers of offspring with certain traits; modify the algorithm to represent a different trait. https://www.birdbraintechnologies.com/hummingbirdduo/teach/project/genetics-simulation/ https://fastplants.org/resources/gene_simulations.php		
2.CCP.4	Evaluate how computer programs can	2.NBT.A.4			2.10 Describe the

2.CCP.5	Create or revise a computational artifact *(a*				2.07, 2.11, 2.13
2.CCP.6	Define a problem or task, decompose it into			2.FL.VA.7a.ii	
2.CCP.7	Collaboratively, students can build				
2.CCP.8	Compare positive and negative impacts				2.26
3.CCP.1	Recognize and understand that a series of				
3.CCP.2	Describe how and why information is broken up and travels in packets (collections of data).		2.PS.4.2 . . . Signals over a distance By analogy: 3.PS1.1 . . . matter . . . particles to small to be seen	By analogy: 3.RL.CS.5 Refer to parts of stories, dramas and poems, using terms such as chapter, scene, and stanza . . .	
3.CCP.3	Identify and determine the purpose of a	3.OA.D.8 Solve . . .			
3.CCP.4	Using a block of code or script from a			*The Poky Little Puppy*	
3.CCP.5	Using a block of code or script from a				
3.CCP.6	Using a block of code or script from a	3.NBT.A.2		Discussion question:	
3.CCP.7	Describe ways that programs and/or			3.18 Analyze how	
4.CCP.1	Recognize the input and output devices along				
4.CCP.2	Demonstrate how information is broken up				
4.CCP.3	Using a block of code or script from an				
4.CCP.4	Construct an algorithm to solve a problem	4.OA.C.5 Generate a	Discuss steps in an		
4.CCP.5	Using a block of code or script that has been				
4.CCP.6	Use existing code and identify sections of				
4.CCP.7	Describe ways that hardware and software		4.PS4.3 . . . computers		
5.CCP.1	Identify and describe the role of various input				
5.CCP.2	Investigate and trace a bundle of information				
5.CCP.3	Decompose (break down) complex real-	5.NF.B.6 Solve Solve			
5.CCP.4	Create an algorithm which includes control				

5.CCP.5	Decompose complex code into subsections or subprograms for reuse into other programs.				5.35 https://tnsoshistory.com/chapter3 Notice how TN's constitutional convention reused sections of the U.S. and NC constitutions.
5.CCP.6	Decompose a piece of code with the intent to			4.FL.SC.6.e Proofread	
5.CCP.7	Formulate alternative uses for software and				
CCP.1	Identify the advantages, disadvantages and unintended				How high-frequency
CCP.2	Analyze the relationship between human and computer				Discuss how this process
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				
CCP.6	Define protocols in relation to a set of rules.		PEMDAS order of		Compore computer
CCP.7	Construct protocols that can be used to share			protocol for	
CCP.8	Compare the relative strengths and weaknesses of				
CCP.9	Create models of networks that include packets and		Determine network		
CCP.10	Identify steps to ensure security measures are in place				
CCP.11	Create cyphers to encrypt data that can be transferred	Encryption algorithms -			Cracking the Enigma
CCP.12	Explain how encryption can be used to safeguard data				
CCP.13	Evaluate the accuracy and precision of various forms				
CCP.14	Identify and define the limiting factors to specific				
CCP.15	Describe how different formats of stored data represent tradeoffs between quality and size.		Record quality vs CD quality?		
CCP.16	Represent data using different encoding schemes, such as binary, Unicode, Morse code, shorthand, student created codes.			Create custom secret language?	
CCP.17	Explain the processes used to collect, transform, and				

CCP.18	Revise variables and constants in computational		ECO.LS2.16		
CCP.19	Solicit and integrate peer feedback as appropriate to				
CCP.20	Compare different algorithms that may be used to	Compare different			
CCP.21	Provide proper attribution when code is borrowed or				
CCP.22	Interpret the flow of execution of algorithms and				
CCP.23	Design, develop, and present computational artifacts				Work into a larger class
CCP.24	Develop programs, both independently and				
CCP.25	Identify the purpose of variables in relation to	Ties nicely into algebra			
CCP.26	Create variables that represent different types of data				
CCP.27	Define and use procedures that hide the complexity of				
CCP.28	Decompose a problem into parts and create solutions		Prompt students to		
CCP.29	Use an iterative design process *(e.g., define the*				
CCP.30	Analyze the positive and negative impacts of				Survey of current events
CCP.31	Recognize there are tradeoffs in computing.				
CCP.32	Explain how social interactions can allow for multiple				
CCP.33	Demonstrate an understanding of digital security				