

Rhode Island Standards Alignment with CodeX Curriculum

	Unit 1	Unit 2	Unit 3
Computational Thinking and Programming			
1B-CTA-1 Compare and refine multiple algorithms for the same task and determine which is more appropriate to complete the task.			
1B-CTV-1 Create programs that use variables	[1]		
1B-CTD-1 Identify real-world examples of data structures and data types.	[2]		
1B-CTC-1 Create programs that combine sequences, loops, conditionals, and/or events.	[3]		
1B-CTM-1 Continually decompose problems into smaller subtasks until each subtask is a manageable set of basic operations.			
1B-CTM-2 Create computational artifacts by incorporating existing modules into one's own work to solve a problem.	[4]		
1B-CTCD-1 Use an iterative process to plan the development of a computational artifact by including others' perspectives and considering user preferences.			
1B-CTCD-2 Debug errors in an algorithm or program that includes sequences and simple loops.	[5]		
1B-CTCD-3 Describe steps taken and choices made during the process of creating a computational artifact.	[6]		
Computing Systems and Networks			
1B-CSN-H-1 Describe how people interact with the various parts of computing systems to accomplish tasks			
1B-CSN-HS-1 Model how computer hardware and software work together as a system to accomplish tasks.	[7]		
1B-CSN-T-1 Determine potential solutions to solve simple hardware and software problems using common troubleshooting strategies.	[8]		
1B-CSN-N-1 Model how information is broken down into smaller pieces of data, transmitted as packets through multiple devices over networks and the Internet, and reassembled at the destination.			
Cybersecurity			
1B-CY-R-1 Describe the risks of sharing personal information, on websites or other public forums.			
1B-CY-R-2 Describe ways personal information can be obtained digitally.			
1B-CY-R-3 Describe the risks of others using one's personal resources or devices.			
1B-CY-S-1 Explain individual actions that protect personal electronic information and devices.			
1B-CY-RP-1 Identify and describe unusual data or behaviors of applications and devices that should be reported to a responsible adult.			
Data and Analysis			
1B-DA-CVT-1 Organize and present collected data to highlight relationships and support a claim.			
1B-DA-IM-1 Use data to highlight or propose cause-and-effect relationships, predict outcomes, or communicate an idea.			
1B-DA-ST-1 Store, copy, search, retrieve, modify, and delete data using a computing device.			
Digital Literacy			
1B-DL-CU-1 Use software tools to create and share multimedia artifacts			
1B-DL-SDI-1 Conduct and refine multi-criteria searches over digital information.			
1B-DL-US-1 Describe the different high-level tasks that are common to software tools that students use.			
Responsible Computing and Society			
1B-RC-CU-1 Compare and contrast computing technologies that have changed the world, and express how those technologies influence, and are influenced by, cultural practices.			
1B-RC-CU-2 Identify ways to improve the accessibility and usability of technology products for the diverse needs and wants of users.			
1B-RC-SLE-1 Incorporate public domain or creative commons media into a digital artifact, and refrain from copying or using material created by others without permission.			
1B-RC-SI-1 Seek diverse perspectives for the purpose of improving computational artifacts.			

Rhode Island Standards Alignment with CodeX Curriculum			
	Unit 1	Unit 2	Unit 3
Computational Thinking and Programming			
2-CT-A-1 Use diagrams and/or pseudocode to plan, analyze, solve and/or code complex problems as algorithms.	[9]		
2-CT-V-1 Create clearly named variables that represent different data. Perform operations on data stored in variables.	[10]		
2-CT-D-1 Organize data into an appropriate data structure in a program.	[11]		
2-CT-C-1 Design programs that combine control structures, including nested loops and compound conditionals.	[12]		
2-CT-M-1 Decompose computational problems to facilitate the design and implementation of programs			
2-CT-M-2 Create procedures with parameters to organize code and make it easier to reuse.	[13]		
2-CT-CD-1 Seek and incorporate feedback from team members and users to refine a solution that meets user needs.			
2-CT-CD-2 Test and debug a program to ensure it runs as intended.	[14]		
2-CT-CD-3 Describe choices made during development of computational artifacts.	[15]		
Computing Systems and Networks			
2-CSN-H-1 Identify improvements to the design of computing devices, based on an analysis of how users interact with the devices.			
2-CSN-HS-1 Design projects that combine hardware and software components to collect and use data to perform a function.		[16]	
2-CSN-T-1 Identify and fix problems with computing devices and their components using a systematic troubleshooting method or guide.	[17]		
2-CSN-N-1 Model the role of protocols in transmitting data across networks and the Internet.			
Cybersecurity			
2-CY-R-1 Describe tradeoffs between allowing information to be public and keeping information private and secure.			
2-CY-R-2 Describe social engineering attacks and the potential risks associated with them.			
2-CY-R-3 Describe risks of using free and open services.			
2-CY-S-1 Explain physical and digital security measures that protect electronic information.			
2-CY-S-2 Demonstrate how multiple methods of encryption provide secure transmission of information.			
2-CY-RP-1 Describe which actions to take and not to take when an application or device reports a problem or behaves unexpectedly.			
Data and Analysis			
2-DA-CVT-1 Collect data using computational tools or online sources and transform the data to make it more useful and reliable.			
2-DA-IM-1 Create and refine computational models based on generated or gathered data.			
2-DA-IM-2 Discuss potential visible biases that could exist in a dataset and how these biases could affect analysis conclusions.			
2-DA-ST-1 Store, retrieve, and share data to collaborate, using a cloud-based system.			
2-DA-ST-2 Describe various low-level data transformations and identify which result in a loss of information.			
Digital Literacy			
2-DL-CU-1 Use software tools to create artifacts that engage users over time.			
2-DL-SDI-1 Conduct searches over multiple types of digital information.			
2-DL-US-1 Describe the different formats of software components that support common tasks in software tools.			
Responsible Computing and Society			
2-RC-CU-1 Compare and contrast tradeoffs associated with computing technologies that affect people's everyday activities and career options.			
2-RC-CU-2 Discuss issues of bias and accessibility in the design of existing technologies.			
2-RC-SLE-1 Discuss how laws control use and access to intellectual property, and mandate broad access to information technologies.			
2-RC-SI-1 Collaborate and strategize with many online contributors when creating a computational or digital artifact.			

Rhode Island Standards Alignment with CodeX Curriculum			
	Unit 1	Unit 2	Unit 3
Computational Thinking and Programming			
3-CT-A-1 Create computational artifacts that use algorithms to solve computational problems by leveraging prior knowledge and personal interests.	[18]		
3-CT-V-1 Explain the role of a variable within a program, and the scope in which its name and value can be used.	[19]		
3-CT-D-1 Create a program that processes a collection of data.			[20]
3-CT-C-1 Create and justify the selection of specific control structures when tradeoffs involve code organization, readability, and program performance and explain the benefits and drawbacks of choices made.			
3-CT-M-1 Identify existing computational artifacts that can be used for the subtasks of a decomposed problem.			
3-CT-M-2 Create computational artifacts by incorporating predefined procedures, self-defined procedures and external artifacts.			
3-CT-CD-1 Systematically design and implement computational artifacts for targeted audiences by incorporating feedback from users.			
3-CT-CD-2 Systematically test and refine programs using a range of test cases.			
3-CT-CD-3 Document computational artifacts in order to make them easier to follow, test, and debug.	[21]		
Computing Systems and Networks			
3-CSN-H-1 Analyze a computing system and explain how abstractions simplify the underlying implementation details embedded in everyday objects.			
3-CSN-HS-1 Compare levels of abstraction and interactions between application software, system software, and hardware layers.			
3-CSN-T-1 Develop and communicate troubleshooting strategies others can use to identify and fix errors.			
3-CSN-N-1 Identify the various elements of a network and describe how they function and interact to transfer information.			
Cybersecurity			
3-CY-R-1 Explain the privacy concerns related to the collection and generation of data through automated processes that may not be evident to users.			
3-CY-R-2 Analyze an existing or proposed application to identify the potential ways it could be used to obtain sensitive information.			
3-CY-R-3 Explain how the digital security of an organization may be affected by the actions of its employees.			
3-CY-S-1 Recommend security measures to address various scenarios based on factors such as efficiency, feasibility, and ethical impacts.			
3-CY-S-2 Explain tradeoffs when selecting and implementing cybersecurity recommendations.			
3-CY-RP-1 Describe the appropriate actions to take in response to detected security breaches.			
Data and Analysis			
3-DA-CVT-1 Select appropriate data-collection tools and presentation techniques for different types of data.			
3-DA-IM-1 Create computational models that represent the relationships among different elements of data collected from a phenomenon or process.			
3-DA-IM-2 Discuss potential hidden biases that could be introduced while collecting a dataset and how these biases could affect analysis conclusions.			
3-DA-IM-3 Evaluate the ability of models and simulations to test and support the refinement of hypotheses.			
3-DA-ST-1 Explain tradeoffs between storing data locally or in central, cloud-based systems.			
3-DA-ST-2 Translate data for various real-world phenomena, such as characters, numbers, and images, into bits.			
Digital Literacy			
3-DL-CU-1 Select appropriate software tools or resources to create a complex artifact or solve a problem.		[22]	
3-DL-SDI-1 Decompose a complex problem into multiple questions, identify which can be explored through digital sources, and synthesize query results using a variety of software tools.			
3-DL-US-1 Describe different kinds of computations that software tools perform to tailor a system to individual users.			
Responsible Computing and Society			
3-RC-CU-1 Evaluate the ways computing impacts personal, ethical, social, economic, and cultural practices.			
3-RC-CU-2 Design and analyze computational artifacts to reduce bias and equity deficits.			
3-RC-CU-3 Evaluate the impact of equity, access, and influence on the distribution of computing resources in a global society.			
3-RC-SLE-1 Evaluate the impact of intellectual property laws on the use of digital information.			
3-RC-SLE-2 Evaluate the social and economic implications of privacy and free speech in the context of safety, law, or ethics.			
3-RC-SI-1 Use tools and methods for collaboration on a project to increase connectivity between people in different cultures and career fields.			

- [1] 3.8 begins the use of variables
- [2] Mission 4 introduces data types
- [3] These all begin in Mission 4
- [4] All missions use libraries and explain the use of any new ones that are introduced
- [5] 3.5 introduces the debugger
- [6] This is the pseudocode that is introduced in the teachers' manual
- [7] Mission 2 introduces this and any time a new sensor or button is introduced it is discussed again.
- [8] These are all of our missions
- [9] Flowcharts and pseudocodes are introduced in the teachers' manual
- [10] 3.8 introduces variables
5.5 discusses descriptive naming of variables
- [11] This is discussed and shown each time a new technique is introduced like where to place global variables and where to import libraries ext.
- [12] Mission 6 introduces but does not discuss nested loops
Mission 9 introduces compound conditionals
- [13] 5.5 introduces the use of comments
7.6 discusses readability and maintenance for code reuse as does 10.6
- [14] 3.5 introduces the use of the debugger
- [15] 5.5 introduces the use of comments to explain code
- [16] All of our missions use a combination of hardware and software, most of our missions towards the end use sensors to collect and use data.
- [17] Mission 2 and the teachers' manual discuss troubleshooting techniques
- [18] These can be the remixes depending on the rubric the teacher uses
- [19] 3.8 introduces the use of variables
5.5 discusses descriptive naming of variables

[20] Mission 10 utilizes the electronic clock circuits' data

[21] 5.5 introduces the use of comments

[22] These are our remixes