Rhode Island Computer Science

Grades 6–8

Computational Thinking & Programming

Algorithms

1 Use diagrams and/or pseudocode to plan, analyze, solve and/or code complex problems as algorithms. 2-CT-A-1

Variables

1 Create clearly named variables that represent different data. Perform operations on data stored in variables. 2-CT-V-1

Data Structures and Data Types

1 Organize data into an appropriate data structure in a program. 2-CT-D-1

Control Structures

1 Design programs that combine control structures, including nested loops and compound conditionals. 2-CT-C-1

Modularity

1 Decompose computational problems to facilitate the design and implementation of programs. 2-CT-M-1

2 Create procedures with parameters to organize code and make it easier to reuse. 2-CT-M-2

Computational Design

1 Seek and incorporate feedback from team members and users to refine a solution that meets user needs. 2-CT-CD-1

2 Test and debug a program to ensure it runs as intended. 2-CT-CD-2

3 Describe choices made during development of computational artifacts. 2-CT-CD-3

Computing Systems & Networks

Human-Computer Interfaces

1 Identify improvements to the design of computing devices, based on an analysis of how users interact with the devices. 2-CSN-H-1

Hardware and Software

1 Design projects that combine hardware and software components to collect and use data to perform a function. 2-CSN-HS-1

Skip to main content

oubleshooting

1 Identify and fix problems with computing devices and their components using a systematic troubleshooting method or guide. 2-CSN-T-1

Networks and the Internet

1 Model the role of protocols in transmitting data across networks and the Internet. 2-CSN-N-1

Cybersecurity

Risks

1 Describe tradeoffs between allowing information to be public and keeping information private and secure. 2-CY-R-1

2 Describe social engineering attacks and the potential risks associated with them. 2-CY-R-2

3 Describe risks of using free and open services. 2-CY-R-3

Safeguards

1 Explain physical and digital security measures that protect electronic information. 2-CY-S-1

2 Demonstrate how multiple methods of encryption provide secure transmission of information. 2-CY-S-2

Response

1 Describe which actions to take and not to take when an application or device reports a problem or behaves unexpectedly. 2-CY-RP-1

Data & Analysis

Collection, Visualization, Transformation

1 Collect data using computational tools or online sources and transform the data to make it more useful and reliable. 2-DA-CVT-1

Inferences and Models

1 Create and refine computational models based on generated or gathered data. 2-DA-IM-1

2 Discuss potential visible biases that could exist in a dataset and how these biases could affect analysis conclusions. 2-DA-IM-2

Storage

1 Store, retrieve, and share data to collaborate, using a cloud-based system. 2-DA-ST-1

2 Describe various low-level data transformations and identify which result in a loss of information 2-DA-ST-2

Digital Literacy

Skip to main content

Creation and Use

Use software tools to create artifacts that engage users over time 2-DL-CU-1

Searching Digital Information

1 Conduct searches over multiple types of digital information. 2-DL-SDI-1

Understanding Software Tools

1 Describe the different formats of software components that support common tasks in software tools. 2-DL-US-1

Responsible Computing & Society

Culture

1 Compare and contrast tradeoffs associated with computing technologies that affect people's everyday activities and career options. 2-RC-CU-1

2 Discuss issues of bias and accessibility in the design of existing technologies. 2-RC-CU-2

Safety, Law & Ethics

1 Discuss how laws control use and access to intellectual property, and mandate broad access to information technologies. 2-RC-SLE-1

Social Interactions

1 Collaborate and strategize with many online contributors when creating a computational or digital artifact. 2-RC-SI-1