

Grades 6-8

Computing Systems

Devices

- 1 Design modifications to computing devices in order to improve the ways users interact with the devices. [6-8.CS.1](#)

Hardware & Software

- 2 Design a project that combines hardware and software components to collect and exchange data. [6-8.CS.2](#)

Troubleshooting

- 3 Systematically apply troubleshooting strategies to identify and resolve hardware and software problems in computing systems. [6-8.CS.3](#)

Networks & the Internet

Network Communication & Organization

- 4 Model the role of protocols in transmitting data across networks and the internet. [6-8.NI.4](#)

Cybersecurity

- 5 Explain potential security threats and security measures to mitigate threats. [6-8.NI.5](#)
- 6 Apply multiple methods of information protection to model the secure transmission of information. [6-8.NI.6](#)

Data & Analysis

Storage

- 7 Represent data in multiple ways. [6-8.DA.7](#)

Collection Visualization & Transformation

- 8 Collect data using computational tools and transform the data to make it more useful. [6-8.DA.8](#)

Inference & Models

- 9 Test and analyze the effects of changing variables while using computational models. [6-8.DA.9](#)
-

Algorithms & Programming

Algorithms

- 10 Use flowcharts and/or pseudocode to design and illustrate algorithms that solve complex problems. 6-8.AP.10
-

Variables

- 11 Create clearly named variables that store data, and perform operations on their contents. 6-8.AP.11
-

Control

- 12 Design and iteratively develop programs that combine control structures and use compound conditions. 6-8.AP.12
-

Modularity

- 13 Decompose problems and subproblems into parts to facilitate the design, implementation, and review of programs. 6-8.AP.13
- 14 Create procedures with parameters to organize code and make it easier to reuse. 6-8.AP.14
-

Program Development

- 15 Seek and incorporate feedback from team members and users to refine a solution that meets user needs. 6-8.AP.15
- 16 Incorporate existing code, media, and libraries into original programs, and give attribution. 6-8.AP.16
- 17 Systematically test and refine programs using a range of test cases. 6-8.AP.17
- 18 Distribute tasks and maintain a project timeline when collaboratively developing computational artifacts. 6-8.AP.18
- 19 Document programs in order to make them easier to use, read, test, and debug. 6-8.AP.19
-

Impacts of Computing

Culture

- 20 Compare tradeoffs associated with computing technologies that affect people's everyday activities and career options. 6-8.IC.20
- 21 Discuss issues of bias and accessibility in the design of existing technologies. 6-8.IC.21
-

Social Interactions

- 22 Collaborate with many contributors when creating a computational artifact. 6-8.IC.22
-

Safety Law & Ethics

- 23 Compare tradeoffs associated with licenses for computational artifacts to balance the protection of the creators' rights and the ability for others to use and modify the artifacts. 6-8.IC.23

24 Compare tradeoffs between allowing information to be public and keeping information private and secure. 6-8.IC.24