

Grades 9-12 Specialty

Computing Systems

Devices

- 1 Illustrate ways computing systems implement logic through hardware components. [9-12S.CS.1](#)

Hardware & Software

- 2 Categorize and describe the different functions of operating system software. [9-12S.CS.2](#)

Networks & the Internet

Network Communication & Organization

- 3 Examine the scalability and reliability of networks, by describing the relationship between routers, switches, servers, topology, and addressing. [9-12S.NI.3](#)
- 4 Explain how the characteristics of the internet influence the systems developed on it. [9-12S.NI.4](#)

Cybersecurity

- 5 Develop solutions to security threats. [9-12S.NI.5](#)
- 6 Analyze cryptographic techniques to model the secure transmission of information. [9-12S.NI.6](#)

Data & Analysis

Collection Visualization & Transformation

- 7 Select and use data collection tools and techniques to generate data sets. [9-12S.DA.7](#)
- 8 Use data analysis tools and techniques to identify patterns in data representing complex systems. [9-12S.DA.8](#)

Inference & Models

- 9 Evaluate the ability of models and simulations to test and support the refinement of hypotheses. [9-12S.DA.9](#)

Algorithms & Programming

Algorithms

- 10 Describe how artificial intelligence drives many software and physical systems. [9-12S.AP.10](#)
- 11 Implement an algorithm that uses artificial intelligence to overcome a simple challenge. [9-12S.AP.11](#)

- 12 Implement searching and sorting algorithms to solve computational problems. 9-12S.AP.12
- 13 Evaluate algorithms in terms of their efficiency. 9-12S.AP.13

Variables

- 14 Compare and contrast fundamental data structures and their uses. 9-12S.AP.14

Control

- 15 Demonstrate the flow of execution of a recursive algorithm. 9-12S.AP.15

Modularity

- 16 Analyze a large-scale computational problem and identify generalizable patterns or problem components that can be applied to a solution. 9-12S.AP.16
- 17 Construct solutions to problems using student-created components, such as procedures, modules, and/or objects. 9-12S.AP.17
- 18 Demonstrate code reuse by creating programming solutions using libraries and APIs. 9-12S.AP.18

Program Development

- 19 Plan and develop programs for broad audiences using a specific software life cycle process. 9-12S.AP.19
- 20 Develop programs for multiple computing platforms. 9-12S.AP.20
- 21 Identify and fix security issues that might compromise computer programs. 9-12S.AP.21
- 22 Develop and use a series of test cases to verify that a program performs according to its design specifications. 9-12S.AP.22
- 23 Modify an existing program to add additional functionality and discuss intended and unintended implications. 9-12S.AP.23
- 24 Evaluate key qualities of a program through a process such as a code review. 9-12S.AP.24
- 25 Use version control systems, integrated development environments (IDEs), and collaborative tools and practices (e.g., code documentation) while developing software within a group. 9-12S.AP.25
- 26 Compare multiple programming languages, and discuss how their features make them suitable for solving different types of problems. 9-12S.AP.26

Impacts of Computing

Culture

- 27 Evaluate computational artifacts with regard to improving their beneficial effects and reducing harmful effects on society. 9-12S.IC.27
- 28 Evaluate how computational innovations that have revolutionized aspects of our culture might evolve. 9-12S.IC.28

- 29 Evaluate the impact of equity, access, and influence on the distribution of computing resources in a global society. 9-12S.IC.29
-

Safety Law & Ethics

- 30 Debate laws and regulations that impact the development and use of software. 9-12S.IC.30