

Grades 11-12

Computing Systems

- 1 Devices: Compare the characteristics and uses of traditional and emerging computing devices and systems.** 11-12.CS.01

- 2 Hardware and Software: Categorize the roles of operating system software.** 11-12.CS.02

- 3 Troubleshooting: Illustrate ways computing systems implement logic, input, and output through hardware components.** 11-12.CS.03

Networks and the Internet

- 4 Network Communication and Organization: Describe the issues that impact network functionality (e.g., bandwidth, load, delay, topology).** 11-12.NI.04

- 5 Cybersecurity: Compare ways software developers protect devices and information from unauthorized access.** 11-12.NI.05

Data and Analysis

- 6 Collection, Visualization, and Transformation: Use data analysis tools and techniques to identify patterns in data representing complex systems.** 11-12.DA.06

- 7 Collection, Visualization, and Transformation: Select data collection tools and techniques to generate data sets that support a claim or communicate information.** 11-12.DA.07

- 8 Collection, Visualization, and Transformation: Analyze the ways in which automated data collection is utilized in society.** 11-12.DA.08

- 9 Interference and Models: Evaluate the ability of models and simulations to test and support the refinement of hypotheses.** 11-12.DA.09

Algorithms and Programming

- 10 Algorithms: Describe how artificial intelligence drives many software and physical systems.** 11-12.AP.10

- 11 Algorithms: Implement an artificial intelligence algorithm to play a game against a human opponent or solve a problem.** 11-12.AP.11

- 12 Algorithms: Use and adapt classic algorithms to solve computational problems.** 11-12.AP.12

- 13 Algorithms: Evaluate algorithms in terms of their efficiency, correctness, and clarity.** 11-12.AP.13

-
- 14 Variables: Compare and contrast fundamental data structures and their uses.** 11-12.AP.14
-
- 15 Control: Illustrate the flow of execution of a recursive algorithm.** 11-12.AP.15
-
- 16 Modularity: Construct solutions to problems using student-created components, such as procedures, modules, or objects.** 11-12.AP.16
-
- 17 Modularity: Analyze a large-scale computational problem and identify generalizable patterns that can be applied to a solution.** 11-12.AP.17
-
- 18 Modularity: Demonstrate code reuse by creating programming solutions using libraries and application programming interfaces.** 11-12.AP.18
-
- 19 Program Development: Plan and develop programs for broad audiences using a software life cycle process.** 11-12.AP.19
-
- 20 Program Development: Demonstrate conversion of source code into machine code using compilers or interpreters.** 11-12.AP.20
-
- 21 Program Development: Explain security issues that might lead to compromised computer programs.** 11-12.AP.21
-
- 22 Program Development: Develop programs for multiple computing platforms.** 11-12.AP.22
-
- 23 Program Development: Use version control systems, integrated development environments, and collaborative tools and practices (code documentation) in a group software project.** 11-12.AP.23
-
- 24 Program Development: Develop and use a series of test cases to verify that a program performs according to its design specifications.** 11-12.AP.24
-
- 25 Program Development: Discuss social, economic, and ethical consequences of malfunctioning software and software updates.** 11-12.AP.25
-
- 26 Program Development: Modify an existing program to add additional functionality and discuss intended and unintended implications (e.g., breaking other functionality).** 11-12.AP.26
-
- 27 Program Development: Evaluate key qualities of a program through a process such as a code review.** 11-12.AP.27
-
- 28 Program Development: Compare multiple programming languages and discuss how their features make them suitable for solving different types of problems.** 11-12.AP.28
-
- 29 Culture: Evaluate computational artifacts to maximize their beneficial effects and minimize harmful effects on society.** 11-12.IC.29

Impacts of Computing

30 Culture: Evaluate the impact of equity, access, and influence on the distribution of computing resources in a global society. 11-12.IC.30

31 Culture: Predict how computational innovations that have revolutionized aspects of our culture might evolve. 11-12.IC.31

32 Safety Law and Ethics: Debate laws and regulations that impact the development and use of software. 11-12.IC.32

Emerging and Future Technologies

A Explain that the field of emerging technologies will be evolving and rapidly growing. 11-12.ET.A

B Compare existing and emerging technologies, ideas, and concepts. 11-12.ET.B

C Describe how emerging technologies are influencing current events at a local and global scale. 11-12.ET.C

D Predict the positive and negative societal, cultural, and economic impacts that emerging and future technologies may generate. 11-12.ET.D

E Create new or original work by applying emerging technologies. 11-12.ET.E
