

NETWORKING / ADVANCED NETWORKING

As aligned to:

South Carolina Computer Science High School Process and Content Standards

The South Carolina Computer Science and Digital Literacy Process Standards should be integrated into every grade level within the South Carolina Computer Science and Digital Literacy Content Standards. Because the Process Standards drive the pedagogical component of teaching and serve as the means by which students should demonstrate understanding of the content standards, the process standards must be incorporated as an integral part of overall student expectations when assessing content understanding.

A computer science literate student can:

1. Foster an inclusive computing culture.
 - a. Recognize that equitable access to computing benefits society as a whole.
 - b. Consider others' perspectives as well as one's own perspective when developing computational solutions.
 - c. Consider the needs of a variety of end users regarding accessibility and usability.
2. Collaborate around computing.
 - a. Select appropriate technological tools that can be used to collaborate on a project.
 - b. Collaborate productively with individuals of varying perspectives, skills, and backgrounds.
 - c. Set and implement equitable expectations and workloads when working in teams.
 - d. Integrate constructive feedback while working in teams.
3. Recognize, define, and analyze computational problems.
 - a. Recognize when it is appropriate to solve a problem computationally.
 - b. Make sense of computational problems and persevere in solving them.
 - c. Relate computational problems to prior knowledge.
 - d. Recognize that there may be multiple approaches to solving a problem.
 - e. Approach problem solving iteratively, using a cyclical process.
4. Create, test, and refine computational artifacts.
 - a. Consider the purpose of computational artifacts for practical use, personal expression, and/or societal impact.
 - b. Recognize when to use the same solution for multiple problems.
 - c. Test computational artifacts systematically by considering multiple scenarios and using test cases.
 - d. Approach troubleshooting systematically.
 - e. Consider performance, reliability, usability, and accessibility when evaluating and refining computational artifacts.
5. Communicate about computing.

- a. Select and use appropriate technological tools to convey solutions to computing problems.
- b. Communicate about computational processes and solutions using appropriate terminology consistent with the intended audience and purpose.
- c. Articulate ideas responsibly by observing intellectual property rights and giving appropriate attribution.

South Carolina Computer Science High School Content Standards

Computing Systems

Standard 1: Examine how hardware and software contribute to computing devices solving relevant problems.

HS3.CS.1.1

Recommend modifications for existing computing devices and software to improve functionality for end users.

HS4.CS.1.1

Develop a solution to a given problem using appropriate hardware and software (e.g., sensor devices, Wi-Fi capabilities, specialized displays, runtime modules, operating systems, application programming interfaces (APIs)).

HS3.CS.1.2

Justify hardware and software selections for specific applications by evaluating the components (e.g., databases, sensors, application programming interfaces (APIs)) of various computing devices (e.g., desktops, laptops, tablets, smartphones, specialized devices like global positioning systems (GPSs)).

HS4.CS.1.2

Cite evidence of how selecting appropriate hardware and software components enhances user interfaces to provide better solutions for real-world problems.

HS1.CS.1.3

Compare and contrast the strengths and weaknesses of various popular operating systems, (e.g., Windows, Windows Server, OS-X, Android, iOS, UNIX, Linux).

HS2.CS.1.3

Analyze the roles of operating system software components (e.g., memory management, data storage and retrieval, process management, access control) in a specific solution (CSTA, 2017).

HS3.CS.1.3

Investigate how the components of an operating system provide simplifying abstractions for a user or developer (e.g., the basic input-output system (BIOS), user interface, memory management, process management, file systems, devices, drivers).

HS4.CS.1.3

Defend the choice of an appropriate operating system based on the requirements of a given computer system or project.

Standard 2: Troubleshoot common hardware and software problems.

HS1.CS.2.1

Interpret various types of error messages from various sources (e.g., operating systems, applications, application programming interfaces (APIs)) to assist in solving common computer malfunctions.

HS2.CS.2.1

Research credible sources of information that can be used for complex troubleshooting strategies (e.g., modifying system settings, correcting connectivity problems).

HS3.CS.2.1

Solve common computer malfunctions or describe the problem accurately, using technical vocabulary, so that others can solve it.

HS4.CS.2.1

Develop guidelines that convey systematic troubleshooting strategies that others can use to identify and fix errors (CSTA, 2017).

Networks and the Internet

Standard 1: Evaluate data transmission across networks, including the Internet.

HS1.NI.1.1

Describe how hardware, software, and protocols work together for transmitting data across networks.

HS2.NI.1.1

Compare and contrast common network protocols (e.g., Internet Protocol (IP), File Transfer Protocol (FTP), Transmission Control Protocol (TCP), Domain Name System (DNS), Hypertext Transfer Protocol (HTTP), Hypertext Transfer Protocol with Secure Sockets Layer (HTTPS), Simple Mail Transfer Protocol (SMTP), Post Office Protocol (POP), Internet Message Access Protocol (IMAP), Telnet, Secure Shell Protocol (SSH)) (Arkansas, 2016).

HS3.NI.1.1

Explain each layer of the Open Systems Interconnect (OSI) Model (Arkansas, 2016).

HS4.NI.1.1

Trace data through the Open Systems Interconnect (OSI) Model.

HS1.NI.1.2
Model the relationship between routers, switches, servers, topology, and addressing (CSTA, 2017).

HS2.NI.1.2

Analyze the tradeoffs and hierarchical structures of various common network topologies (Arkansas, 2016).

HS3.NI.1.2

Design a practical, efficient, and secure network solution (e.g., small office network) (Arkansas, 2016).

HS4.NI.1.2

Evaluate the scalability and reliability of a network (e.g., bandwidth, load, delay, topology) (CSTA, 2017).

Standard 2: Evaluate cybersecurity threats and appropriate security measures across networks.

HS1.NI.2.1

Reflect on case studies or current events in which governments or organizations experienced data leaks or data loss as a result of cybersecurity attacks (CSTA, 2017).

HS2.NI.2.1

Evaluate how sensitive data can be affected by malware and other attacks (e.g., denial-of-service attacks, ransomware, viruses, worms, spyware, phishing) (CSTA, 2017).

HS3.NI.2.1

Research security measures (i.e., hardware, software, and practices that control access to data and systems) to combat a variety of cybersecurity vulnerabilities (CSTA, 2017).

HS4.NI.2.1

Recommend security measures (i.e., hardware, software, and practices that control access to data and systems) to address various scenarios based on factors such as efficiency, feasibility, and ethical impacts (CSTA, 2017).

HS1.NI.2.2

Identify best practices of software development that improve computer security and protect devices and information from unauthorized access (e.g., encryption, authentication strategies, secure coding, safeguarding keys) (CSTA, 2017).

HS2.NI.2.2

Compare and contrast ways software developers protect devices and information from unauthorized access (e.g., encryption, authentication strategies, secure coding, safeguarding keys) (CSTA, 2017).

HS3.NI.2.2

Evaluate various security measures, considering tradeoffs between the usability and security of a computing system. (e.g., a web filter that prevents access to many educational sites but keeps a campus' network safe) (CSTA, 2017).

HS4.NI.2.2

Select and justify cybersecurity recommendations (i.e., hardware, software, and practices that control access to data and systems) appropriate for an intended audience and purpose (CSTA, 2017).

Data and Analysis - None

Algorithms and Programming - None

Impact of Computing

Standard 2: Evaluate the evolving legal and ethical tradeoffs that shape computing practices.

HS4.IC.2.2

Recommend methods to protect digital information in different situations (e.g., traveling to other countries, two-factor authentication, encryption).

