

High School: Computer Science & Mathematics

Computer Systems and Computational Thinking

Evaluate different data representations to solve problems.

- 1 Analyze the various mathematical bases (e.g., binary, decimal, hexadecimal) and convert between them. [CS.M.1](#)
- 2 Describe the relationship between binary and hexadecimal representations. [CS.M.2](#)
- 3 Convert information between various encoding formats (e.g., ASCII, Unicode, hexadecimal, binary). [CS.M.3](#)
- 4 Compare techniques (e.g., sorting, statistics, searching) for analyzing massive data collections. [CS.M.4](#)

Connect the development cycle of algorithm construction to problem-solving.

- 5 Describe how mathematical and statistical functions, sets, and logic are used in computation. [CS.M.5](#)
- 6 Utilize predefined mathematical functions and parameters to divide a complex problem into simpler parts, including parallel processing. [CS.M.6](#)
- 7 Interpret truth tables from basic statements using Boolean operators (AND, OR, XOR, and NOT). [CS.M.7](#)
- 8 Explain ways in which sequence, selection, iteration, and recursion are building blocks of algorithms. [CS.M.8](#)
- 9 Create systems of equations based on real-world situations. [CS.M.9](#)
- 10 Analyze decisions and strategies using probability and statistical concepts. [CS.M.10](#)

Create and evaluate algorithms to solve problems.

- 11 Utilize modeling and simulation techniques to represent and understand natural phenomena. [CS.M.11](#)
- 12 Examine classical algorithms (e.g., searching, sorting, and shortest path). [CS.M.12](#)
- 13 Manipulate formulas and equations and apply them to algorithm development. [CS.M.13](#)
- 14 Apply algorithm analysis and design techniques to solve problems. [CS.M.14](#)
- 15 Write algorithms to solve mathematical problems using formulas, equations, and functions. [CS.M.15](#)
- 16 Implement conditional statements that include if/then, if/then/else, case statements, and Boolean logic, in the design of algorithms. [CS.M.16](#)
- 17 Represent algorithms using flowcharts and pseudocode. [CS.M.17](#)
- 18 Combine standard function types using arithmetic operations. [CS.M.18](#)
- 19 Analyze algorithms for correctness, clarity, and efficiency. [CS.M.19](#)

Programming and Algorithms**Evaluate the use of programming languages to solve problems and develop systems.**

- 20 Compare and contrast computer programming languages and paradigms (e.g., compiled and interpreted languages, procedural and object-oriented paradigms). [CS.M.20](#)
- 21 Diagram the program execution process. [CS.M.21](#)
- 22 Determine the output of a given sample program without the use of a computer. [CS.M.22](#)

Create, test, and use computer programs to solve problems.

- 23 Implement computing applications using the following software development tools and techniques [CS.M.23](#)
 - 1 branching (if, if-else) [CS.M.23.1](#)
 - 2 declare, define, and reference variables [CS.M.23.2](#)
 - 3 lists/arrays [CS.M.23.3](#)
 - 4 looping (for, while, do/while) [CS.M.23.4](#)
 - 5 recursion [CS.M.23.5](#)
 - 6 sequencing [CS.M.23.6](#)
 - 24 Use various debugging and testing methods to ensure program correctness. [CS.M.24](#)
 - 25 Cite evidence to support or refute the correctness of software solutions. [CS.M.25](#)
-

Computers and Communication Devices

Classify electronic devices containing computational processors that execute programs.

- 26 Recognize that computers are devices that execute programs. [CS.M.26](#)
 - 27 Identify a variety of electronic devices (e.g., cell phones, desktops, laptops, vehicles, programmable thermostats, and programmable kitchen appliances) that contain computational processors. [CS.M.27](#)
 - 28 Describe unique features of computers embedded in mobile devices and vehicles. [CS.M.28](#)
 - 29 Investigate the history of computers, identifying contributors and major milestones (e.g., Alan Turing, Charles Babbage, Ada Lovelace, Grace Hopper, analytical machine, ENIAC, IBM PC). [CS.M.29](#)
-

Analyze the relationship between hardware and software.

- 30 Demonstrate an understanding of the relationship between hardware and software. [CS.M.30](#)
 - 31 Develop criteria for purchasing or upgrading computer system hardware. [CS.M.31](#)
 - 32 Describe primary components of computer systems (e.g., input, output, processing, and storage). [CS.M.32](#)
 - 33 Explain multiple levels of hardware and software that support program execution (e.g., compilers, interpreters, operating systems, networks). [CS.M.33](#)
 - 34 Apply strategies for identifying and solving routine hardware problems that occur during everyday computer use. [CS.M.34](#)
-

Describe the major components and functions of networks.

- 35 Describe how the Internet facilitates global communication. [CS.M.35](#)
 - 36 Describe issues that impact network functionality (e.g., latency, bandwidth, firewalls, server capability). [CS.M.36](#)
-

Impacts of Computing

Evaluate appropriate and inappropriate uses of technology.

- 37 Summarize appropriate and inappropriate technological behaviors, including issues of privacy, copyright, security, legalities, and politics. [CS.M.37](#)
- 38 Explore the ramifications of inappropriate uses of technology. [CS.M.38](#)
- 39 Investigate the national and global economic impact of cybercrime. [CS.M.39](#)

Investigate social and ethical issues relating to digital information.

- 40 Discuss accessibility issues (e.g., adaptive technology for special needs individuals, censorship, geographical locations, and economically-disadvantaged populations). [CS.M.40](#)
- 41 Compare the reliability of various online sources. [CS.M.41](#)
- 42 Investigate information ownership topics [CS.M.42](#)
 - 1 access [CS.M.42.1](#)
 - 2 distribution rights [CS.M.42.2](#)
 - 3 hacking [CS.M.42.3](#)
 - 4 licensure [CS.M.42.4](#)
 - 5 open source [CS.M.42.5](#)
 - 6 public domain [CS.M.42.6](#)
 - 7 software piracy [CS.M.42.7](#)
- 43 Describe security and privacy issues that relate to computer networks. [CS.M.43](#)

Will explore security and privacy techniques.

- 44 Explain principles of network security and techniques that protect stored and transmitted data (e.g., encryption, cryptography, and authentication). [CS.M.44](#)