Montana Computer Science

Grades 9-12

When a district incorporates or integrates computer science content into district curriculum or offers a course in computer science, the following skills at each grade level apply

1. **fostering an inclusive computing culture** CS.1.K12
2. **collaborating around computing** CS.2.K12
3. **recognizing and defining computational problems** CS.3.K12
4. **developing and using abstractions** CS.4.K12
5. **creating computational artifacts** CS.5.K12
6. **testing and refining computational artifacts** CS.6.K12
7. **communicating about computing.** CS.7.K12

Computer science algorithms and programming standards for ninth through twelfth grades are that each student will CS.AP.9-12

1. **create prototypes that use algorithms to solve computational problems by leveraging prior student knowledge and personal interests** CS.AP.9-12.1
2. **describe how artificial intelligence drives many software and physical systems** CS.AP.9-12.2
3. **implement an artificial intelligence algorithm to play a game against a human opponent or solve a problem** CS.AP.9-12.3
4. **use and adapt classic algorithms to solve computational problems** CS.AP.9-12.4
5. **evaluate algorithms in terms of their efficiency, correctness, and clarity** CS.AP.9-12.5
6. **use lists to simplify solutions, generalizing computational problems instead of repeatedly using simple variables** CS.AP.9-12.6
7. **compare and contrast fundamental data structures and their uses** CS.AP.9-12.7
8. **justify the selection of specific control structures when tradeoffs involve implementation, readability, and program performance, and explain the benefits and drawbacks of choices made** CS.AP.9-12.8
9. **design and iteratively develop computational artifacts for practical intent, personal expression, or to address a societal issue by using events to initiate instructions** CS.AP.9-12.9

Skip to main content

10 **decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, or objects** CS.AP.9-12.10

11 **create artifacts by using procedures within a program, combinations of data and procedures, or independent but interrelated programs** CS.AP.9-12.11

12 **construct solutions to problems using student-created procedures, modules, or objects** CS.AP.9-12.12

13 **analyze a large-scale computational problem and identify generalizable patterns that can be applied to a solution** CS.AP.9-12.13

14 **demonstrate code reuse by creating programming solutions using libraries and application programming interfaces** CS.AP.9-12.14

15 **systematically design and develop programs for broad audiences by incorporating feedback from users** CS.AP.9-12.15

16 **evaluate and refine computational artifacts to make them more usable and accessible** CS.AP.9-12.16

17 **design and develop computational artifacts working in team roles using collaborative tools** CS.AP.9-12.17

18 **document design decisions using text, graphics, presentations, or demonstrations in the development of complex programs** CS.AP.9-12.18

19 **plan and develop programs for broad audiences using a software life cycle process** CS.AP.9-12.19

20 **explain security issues that might lead to compromised computer programs** CS.AP.9-12.20

21 **develop programs for multiple computing platforms** CS.AP.9-12.21

22 **use version control systems, integrated development environments, and collaborative tools and practices in a group software project** CS.AP.9-12.22

23 **develop and use a series of test cases to verify that a program performs according to its design specifications** CS.AP.9-12.23

24 **modify an existing program to add additional functionality and discuss intended and unintended implications** CS.AP.9-12.24

25 **evaluate key qualities of a program through a process such as a code review** CS.AP.9-12.25

26 **compare multiple programming languages and discuss how their features make them suitable for solving different types of problems.** CS.AP.9-12.26

Skip to main content

Computer science computing systems standards for ninth through twelfth grades are that each student will CS.CS.9-12

1. **explain how abstractions hide the underlying implementation details of computing systems embedded in everyday objects** CS.CS.9-12.1
2. **compare levels of abstraction and interactions between application software, system software, and hardware layers** CS.CS.9-12.2
3. **categorize the roles of operating system software** CS.CS.9-12.3
4. **develop guidelines that convey systematic troubleshooting strategies that others can use to identify and fix errors** CS.CS.9-12.4
5. **illustrate ways computing systems implement logic, input, and output through hardware components.** CS.CS.9-12.5

Computer science data and analysis standards for ninth through twelfth grades are that each student will CS.DA.9-12

1. **create interactive data visualizations using software tools to help others better understand authentic phenomena** CS.DA.9-12.1
2. **use data analysis tools and techniques to identify patterns in data representing complex systems** CS.DA.9-12.2
3. **select data collection tools and techniques to generate data sets that support a claim or communicate information** CS.DA.9-12.3
4. **translate between different bit representations of authentic phenomena, including characters, numbers, and images** CS.DA.9-12.4
5. **evaluate the tradeoffs in how data elements are organized and where data is stored** CS.DA.9-12.5
6. **create computational models that represent the relationships among different elements of data collected from a phenomenon or process** CS.DA.9-12.6
7. **evaluate the ability of models and simulations to test and support the refinement of hypotheses.** CS.DA.9-12.7

Computer science impacts of computing standards for ninth through twelfth grades are that each student will CS.IC.9-12

1. **evaluate the ways computing technologies, globally and locally impact personal, ethical, social, economic, and cultural practices** CS.IC.9-12.1
2. **evaluate the ways computing technologies impact American Indian communities in Montana** CS.IC.9-12.2
3. **test and refine computational artifacts to reduce bias and equity deficits** CS.IC.9-12.3
4. **demonstrate ways a given algorithm applies to problems across disciplines** CS.IC.9-12.4
5. **valuate computational artifacts to maximize their beneficial effects and minimize armful effects on society** CS.IC.9-12.5

Skip to main content

6 evaluate the impact of equity, access, and influence on the distribution of computing resources in a global society, including the impact on American Indians living in urban, rural, and reservation communities CS.IC.9-12.6

7 predict how computational innovations that have revolutionized aspects of our culture might evolve CS.IC.9-12.7

8 use tools and methods to connect and work with others on a project including people in different cultures and career fields CS.IC.9-12.8

9 explain the beneficial and harmful effects that intellectual property laws can have on innovation CS.IC.9-12.9

10 explain the privacy concerns related to the collection and generation of data through automated processes that may not be evident to users CS.IC.9-12.10

11 evaluate the social and economic implications of privacy in the context of safety, law, or ethics CS.IC.9-12.11

12 debate laws and regulations that impact the development and use of software CS.IC.9-12.12

Computer science networks and the internet standards for ninth through twelfth grades are that each student will CS.NI.9-12

1 recommend security measures to address various scenarios based on factors including efficiency, feasibility, and ethical impacts CS.NI.9-12.1

2 explain tradeoffs when selecting and implementing cybersecurity recommendations CS.NI.9-12.2

3 compare ways software developers protect devices and information from unauthorized access CS.NI.9-12.3

4 evaluate the scalability and reliability of networks by describing the relationship between routers, addressing, switches, servers, and topology CS.NI.9-12.4

5 give examples to illustrate how sensitive data can be affected by malware and other attacks CS.NI.9-12.5

6 compare various security measures, considering tradeoffs between the usability and security of a computing system CS.NI.9-12.6

7 discuss the issues that impact functionality CS.NI.9-12.7