Structure

Design Brief for Standards Writers

SUGGESTED CITATION: Computer Science Teachers Association. (2026). *Structure: Design brief for standards writers*. https://csteachers.org/k12standards/research

WestEd.org

Integrating the Pillars

Each standard should explicitly link a subtopic and one or more practices, which are embedded within the Pillars.

Definition of Impacts and Ethics[1]

Key Ideas	**Practice:** Apply ethical principles when developing or evaluating computational tools.
1. Access to and use of computing technologies differs across groups of people (e.g., by socioeconomic status; geographic location; cultural background; differences in ability, gender, race, and ethnicity; and other social identities). **2.** Computing has both positive and negative impacts on society and the environment. **a.** Over-optimism about computing technologies can lead to ethical challenges. **b.** Computing technologies impact people differently depending on their values, socioeconomic status, geographic location, cultural background, abilities, and social identities. **c.** There are laws to regulate the design and use of computing technologies, as well as unsuccessful attempts to enact such laws. **3.** Responsible creation and use of computing technology requires consideration of ethical issues. **a.** Decisions made about the design and use of computing technologies involve consideration of costs, benefits, and trade-offs. **b.** Computing technology should be optimized to perform the task for which it was designed with the least possible harmful impacts on users and the environment.	**1.** Use computing for the public good. **a.** Acknowledge that all people hold an interest in computing. **b.** Respect diversity. **c.** Promote environmental sustainability of computing technologies. **2.** Avoid harm caused by computing technologies, including physical or mental injury, unjustified destruction or disclosure of information, and unjustified damage to property, reputation, or the environment. **a.** Carefully consider the potential impacts of new computing technologies to avoid unintentionally causing harm. **b.** Undo or mitigate harm caused by computing technologies as much as possible. **3.** Be honest and trustworthy. **4.** Be fair and do not discriminate. **a.** Create computing technologies that are as inclusive and accessible as possible. **b.** Do not create technologies that disenfranchise or oppress people. **5.** Respect the work required to produce new ideas, inventions, creative works, and computing artifacts. **a.** Only use others' work with permission. **b.** Give appropriate attribution to others' work. **6.** Respect privacy. **a.** Protect data from unauthorized access and accidental disclosure. **b.** Only collect the minimum amount of information necessary. **c.** Clearly describe what data is being collected, how it is being stored and used, and when it will be disposed of. **d.** Do not use data collected for a specific purpose for another purpose without consent.

[1] Impacts and Ethics should be a distinct subtopic in every topic area that focuses on the key ideas.

Definition of Inclusive Collaboration

Practice: Communicate clearly, using oral, written, and nonverbal methods, and practice intercultural competence.	**Practice:** Actively manage projects and coordinate with others.	**Practice:** Act responsibly and demonstrate accountability to others.
1. Know when to listen and when to speak. 2. Cultivate working relationships with individuals possessing diverse perspectives, skills, and personalities, and aim to build empathy, respect, understanding, and connection. 3. Express curiosity about the perspectives of others and exchange ideas in open-minded ways. 4. Leverage social and cultural differences to create new ideas, increase innovation, and improve the quality of work.	1. Establish common goals and team norms. 2. Leverage team members' strengths to accomplish common goals. 3. Define team roles and create equitable workloads. 4. Value the individual contributions of each team member. 5. Manage time effectively. 6. Document work products and processes.	1. Solicit and incorporate feedback from team members and other interested parties. 2. Participate actively and be reliable and punctual. 3. Assume shared responsibility for collaborative work. 4. Be flexible, helpful, and willing to make necessary compromises to accomplish a common goal. 5. Exhibit honesty and integrity.

Definition of Computational Thinking

Practice: Make sense of problems, test, and troubleshoot.	**Practice:** Look for patterns, develop and use abstractions, and modularize.
1. Analyze computational problems, make conjectures about the sources of errors and possible solutions, and plan a solution pathway rather than simply jumping into a solution attempt. 2. Systematically test computational artifacts by considering all scenarios and using test cases. 3. Use systematic processes to identify and fix errors.	1. Decompose complex problems into manageable subproblems that can integrate existing solutions or procedures. 2. Look for patterns and repetition. Identify and create general methods for performing repetitive tasks. 3. Extract common features from a set of integrated processes or complex phenomena. 4. Evaluate existing functional modules and incorporate them into new designs. 5. Create modules and develop points of interaction between them that can apply to multiple situations to reduce complexity.

Definition of Human-Centered Design[2]

Key Ideas	**Practice:** Apply principles of human-centered design when creating computational artifacts.
1. Human-centered design is an approach to interactive systems that aims to make systems usable and useful by focusing on the users and their needs and requirements, and by applying human factors/ergonomics, and usability knowledge and techniques. This approach enhances effectiveness and efficiency; improves human well-being, user satisfaction, accessibility, and sustainability; and counteracts possible adverse effects of use on human health, safety, and performance. **2.** There are four principles of human-centered design: **a.** *Be people-centered.* In other words, "know your user." Understand and focus on their needs, abilities, and contexts. **b.** *Understand and solve the right problems: the root problems.* Distinguish between the root cause of a problem and the symptoms of a problem. Solve the underlying problem rather than addressing symptoms. **c.** *Think of everything as a system of interconnected parts.* **d.** *Start with small and simple interventions and iterate* rather than rushing into big design solutions. Continually prototype, test, and refine your solutions to make sure they meet people's needs. **3.** Human-centered design is related to humanity-centered design, which focuses on designing systems that consider the entire ecosystem of people, all living things, and the environment.	**1.** Identify and define real-world problems that can be solved computationally, the people impacted by the problems, and any requirements of or constraints on the eventual solutions. **2.** Design and develop computational solutions to real-world problems. Begin by generating several possible solutions and evaluating them to see which one best meets the criteria for resource use and user experience. **3.** Test, refine, and optimize computational solutions. Test the computational solutions with diverse users. Prioritize improvements to the most important features.

[2] The key ideas of Human-Centered Design should appear as a distinct subtopic in at least the Programming topic area.

Comparison of Grade-Level/Grade-Band Structures

WestEd reviewed a large number of standards and frameworks. The table below shows the grade-level/grade-band structure for each.

<table>
<tr><th></th><th>Standards/Frameworks</th><th>PreK</th><th>K</th><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th><th>7</th><th>8</th><th>9–10</th><th>11–12</th></tr>
<tr><td rowspan="5">CS-Specific Standards and Frameworks</td><td>K–12 CS Framework</td><td>N/A</td><td colspan="3">● ● ●</td><td colspan="3">● ● ●</td><td colspan="3">● ● ●</td><td colspan="2">● ● ● ●</td></tr>
<tr><td>2017 K–12 CS Standards</td><td>N/A</td><td colspan="3">● ● ●</td><td colspan="3">● ● ●</td><td colspan="3">● ● ●</td><td>● ●</td><td>● ●</td></tr>
<tr><td>EU Informatics Ref. Framework</td><td colspan="12">Primary, lower secondary, upper secondary</td></tr>
<tr><td>AI4K12</td><td>N/A</td><td colspan="3">● ● ●</td><td colspan="3">● ● ●</td><td colspan="3">● ● ●</td><td colspan="2">● ● ● ●</td></tr>
<tr><td>K–12 Cybersec. Learning Stds.</td><td>N/A</td><td colspan="3">● ● ●</td><td colspan="3">● ● ●</td><td colspan="3">● ● ●</td><td colspan="2">● ● ● ●</td></tr>
<tr><td rowspan="4">Other STEM Standards and Frameworks</td><td>Common Core State Stds.—Math</td><td>N/A</td><td>●</td><td>●</td><td>●</td><td>●</td><td>●</td><td>●</td><td>●</td><td>●</td><td>●</td><td colspan="2">● ● ● ●</td></tr>
<tr><td>GAISE II</td><td colspan="12">Levels A, B, and C; no grade-specific instructional recommendations</td></tr>
<tr><td>Next Generation Science Stds.</td><td>N/A</td><td>●</td><td>●</td><td>●</td><td>●</td><td>●</td><td>●</td><td colspan="3">● ● ●</td><td colspan="2">● ● ● ●</td></tr>
<tr><td>Stds. for Tech. and Eng. Literacy</td><td colspan="4">● ● ● ●</td><td colspan="3">● ● ●</td><td colspan="3">● ● ●</td><td colspan="2">● ● ● ●</td></tr>
<tr><td rowspan="5">Arts and Humanities Standards and Frameworks</td><td>Common Core State Stds.—ELA</td><td>N/A</td><td>●</td><td>●</td><td>●</td><td>●</td><td>●</td><td>●</td><td>●</td><td>●</td><td>●</td><td>● ●</td><td>● ●</td></tr>
<tr><td>Nat'l Curr. Stds. for Social Studies</td><td colspan="6">● ● ● ● ● ●</td><td colspan="4">● ● ● ●</td><td colspan="2">● ● ● ●</td></tr>
<tr><td>National Core Arts Standards</td><td>●</td><td>●</td><td>●</td><td>●</td><td>●</td><td>●</td><td>●</td><td>●</td><td>●</td><td>●</td><td colspan="2">● ● ● ●</td></tr>
<tr><td>Common Career Technical Core</td><td>N/A</td><td colspan="11">● ● ● ● ● ● ● ● ● ● ● ● ●</td></tr>
<tr><td>Social Justice Standards</td><td>N/A</td><td colspan="3">● ● ●</td><td colspan="3">● ● ●</td><td colspan="3">● ● ●</td><td colspan="2">● ● ● ●</td></tr>
</table>

Based on the frequency of different types of structures, we recommend the following for the revised CSTA CS standards:

- PreK standards should be included as part of a grade band (e.g., PreK–2) if they are included at all.
- Elementary school standards (grades K–5) can be written for individual grades or for two grade bands: K–2 and 3–5.
- Middle school standards (grades 6–8) should be banded together.
- High school standards (grades 9–12) should be banded together.

Recommendation to Vary the Structure of the Standards by Different Grade Bands or Topic Areas

The 2017 CSTA K–12 CS Standards are very uniform. For the most part, all concepts and subconcepts are addressed in every grade band, often with exactly one standard per subconcept/grade-band pair. In contrast, the Common Core State Standards for both Mathematics (CCSSM) and English Language Arts (CCSSELA) have variable structures to reflect different content needs for students in different grade bands, though this can create some challenges for curriculum and assessment development.

Topic Structure of the Common Core State Standards for Mathematics and Numbers of Standards per Grade

Domain	K	1	2	3	4	5	6	7	8	HS
Counting and Cardinality	10	-	-	-	-	-	-	-	-	-
Operations and Algebraic Thinking	5	8	4	9	5	3	-	-	-	-
Number and Operations in Base Ten	1	9	11	3	6	9	-	-	-	-
Number and Operations—Fractions	-	-	-	9	14	14	-	-	-	-
Ratio and Proportional Relationships	-	-	-	-	-	-	8	7	-	-
The Number System	-	-	-	-	-	-	15	11	2	-
Expressions and Equations	-	-	-	-	-	-	12	6	13	-
Functions	-	-	-	-	-	-	-	-	5	45
Number and Quantity	-	-	-	-	-	-	-	-	-	32
Algebra	-	-	-	-	-	-	-	-	-	34
Measurement and Data	3	4	10	14	9	10	-	-	-	-
Statistics and Probability	-	-	-	-	-	-	9	13	4	36
Modeling[3]	●	●	●	●	●	●	●	●	●	●
Geometry	6	3	3	2	3	4	4	6	12	45
Total # Standards/Substandards	**25**	**24**	**28**	**37**	**37**	**40**	**48**	**43**	**36**	**192**

[3] Modeling is addressed in Standard for Mathematical Practice 4 and thus should appear in all grade levels. In high school, modeling is not a standalone conceptual category. Instead, specific standards throughout the high school standards are designated as modeling standards.

Topic Structure of the Common Core State Standards for English Language Arts and Numbers of Standards per Grade

Anchor Standards		K	1	2	3	4	5	6	7	8	9–10	11–12
Reading	Foundational Skills	17	19	11	9	6	6	-	-	-	-	-
	Literature	9	9	9	9	9	9	9	9	9	9	9
	Informational Text	10	10	10	10	10	10	10	10	10	10	10
	History/Social Studies[4]	-	-	-	-	-	-	10			10	10
	Science & Technical Subjects	-	-	-	-	-	-	10			10	10
Writing	General	7	7	7	21	26	26	28	28	28	28	28
	History/Social Studies; Science & Tech. Subjects	-	-	-	-	-	-	20			20	19
Speaking and Listening		8	9	9	10	10	10	10	10	10	10	10
Language		21	26	25	31	26	24	22	19	21	18	17
Total # Standards for ELA		**72**	**80**	**71**	**90**	**87**	**85**	**79**	**76**	**78**	**75**	**74**
Total # Standards for Other Subjects		-	-	-	-	-	-	**40**			**40**	**39**

Ways to vary the structure of the standards might include the following:

- For any given topic, write more standards for middle and high school than for elementary school.
- Include subtopics that have standards for only part of the K–12 progression.
  - » *Example:* If the Data and Analysis topic area will include a subtopic about statistical analysis, it should begin in middle school, based on the progressions in the CCSSM.
- Provide alternative structures for content that is not tied to specific grade bands but that must be taught sequentially.
  - » *Example:* Programming is a topic area where students need to progress through fundamentals of programming, such as variables and control structures, regardless of when they begin to program. Consider using a grade-independent level structure (e.g., introductory, intermediate, and advanced) or developing alternative sequences of standards for students who begin programming in different grade bands.

---

[4] Fully grayed-out rows are ELA standards for non-ELA subjects.

Overall Recommendations for Writing Standards

- Standards should generally be one sentence of around 20 words, with one verb.
- Write 10–15 standards per grade level and a total of 150–200 standards.
- Write fewer standards per grade for K–5 than for 6–12.

Standards Should Be Accompanied by Supplementary Material

Generally speaking, standards are descriptions of what students should know and be able to do after instruction. They should be written to allow for flexibility in instruction and assessment. The standards we reviewed varied in how much additional information they included. Writers will have to decide how much additional content to include with each standard, but any additional content should be kept brief, similar to the length of the standard itself.

We recommend that the standards writers develop supplementary material similar to those included for other sets of standards, either as a separate document or as an introduction or appendices to the standards. **At a minimum, we recommend material for each of the following:**

- Defining the topic areas and the learning progression within each topic area
- Defining the Pillars and the learning progression within each Pillar
- Defining the dispositions and the learning progression within each disposition
- Connections to the CCSSM (see Figure 1 for example)
- Connections to the NGSS

Students who demonstrate understanding can:

HS-ETS1-4. **Use a computer simulation to model the impact of proposed solutions to a complex real-world problem with numerous criteria and constraints on interactions within and between systems relevant to the problem.**

The performance expectation above was developed using the following elements from the NRC document *A Framework for K-12 Science Education*:

Science and Engineering Practices	Disciplinary Core Ideas	Crosscutting Concepts
Using Mathematics and Computational Thinking Mathematical and computational thinking in 9-12 builds on K-8 experiences and progresses to using algebraic thinking and analysis, a range of linear and nonlinear functions including trigonometric functions, exponentials and logarithms, and computational tools for statistical analysis to analyze, represent, and model data. Simple computational simulations are created and used based on mathematical models of basic assumptions. • Use mathematical models and/or computer simulations to predict the effects of a design solution on systems and/or the interactions between systems.	**ETS1.B: Developing Possible Solutions** • Both physical models and computers can be used in various ways to aid in the engineering design process. Computers are useful for a variety of purposes, such as running simulations to test different ways of solving a problem or to see which one is most efficient or economical; and in making a persuasive presentation to a client about how a given design will meet his or her needs.	**Systems and System Models** • Models (e.g., physical, mathematical, computer models) can be used to simulate systems and interactions—including energy, matter, and information flows— within and between systems at different scales.

Connections to HS-ETS1.B: Developing Possible Solutions Problems include:
Earth and Space Science: HS-ESS3-2, HS-ESS3-4 **Life Science:** HS-LS2-7, HS-LS4-6

Articulation of DCIs across grade-levels:
MS.ETS1.A ; **MS.ETS1.B** ; **MS.ETS1.C**

Common Core State Standards Connections:
Mathematics -
MP.2 Reason abstractly and quantitatively. *(HS-ETS1-4)*
MP.4 Model with mathematics. (HS-ETS1-4)

Figure 1. *Example of an NGSS standard that includes connections to the CCSSM*

For more information on these design recommendations, go to https://csteachers.org/k12standards/research. See the following sections of *Comparison of CSTA Standards to Other Standards and Frameworks*: Standards for Technological and Engineering Literacy (pp. 38–45), National Curriculum Standards for Social Studies (pp. 48–49), and Standards Structure (pp. 56–64).

Works Consulted

CS-Specific Standards and Frameworks

- K–12 Computer Science Framework: https://k12cs.org/
- 2017 CSTA K–12 Computer Science Standards: https://csteachers.org/k12standards/
- European Union Informatics Reference Framework for School: https://www.informaticsforall.org/the-informatics-reference-framework-for-school-release-february-2022/
- Artificial Intelligence for K–12 (AI4K12) Guidelines: https://ai4k12.org/gradeband-progression-charts/
- K–12 Cybersecurity Learning Standards: https://cyber.org/standards

Other STEM Standards and Frameworks

- Common Core State Standards for Mathematics: https://www.thecorestandards.org/Math/
- PreK–12 Guidelines for Assessment and Instruction in Statistics Education II (GAISE II): https://www.amstat.org/asa/files/pdfs/GAISE/GAISEIIPreK-12_Full.pdf
- Next Generation Science Standards: https://www.nextgenscience.org/standards
- Standards for Technological and Engineering Literacy: https://www.iteea.org/stel

Arts and Humanities Standards and Frameworks

- Common Core State Standards for English Language Arts: https://www.thecorestandards.org/ELA-Literacy/
- National Curriculum Standards for Social Studies: https://www.socialstudies.org/standards/national-curriculum-standards-social-studies
- National Core Arts Standards: https://www.nationalartsstandards.org/
- Common Career Technical Core: https://osse.dc.gov/sites/default/files/dc/sites/osse/publication/attachments/CCTC%20Standards.pdf
- Social Justice Standards: https://www.learningforjustice.org/frameworks/social-justice-standards

