Colorado Academic Standards

ALL STUDENTS ALL STANDARDS

COMPUTER SCIENCE
FALL 2017 ONLINE FEEDBACK

COLORADO
Department of Education

As part of the review and revision of the Colorado Academic Standards (CAS), the Colorado Department of Education invited the public to review and give feedback to proposed revisions to the CAS. These public comments, suggestions, and indications of approval were collected through an online feedback system that was open from October 16, 2017 through December 4, 2017. Below is the public feedback submitted for Computer Science. Only those parts of the standards (Prepared Graduate Statements, Grade Level Expectations, Evidence Outcomes, etc.) that received feedback are listed below, and as such, some PGs/GLEs/EOs are missing from this document. For more information on the Colorado Academic Standards and the 2018 review and revision, visit http://www.cde.state.co.us/standardsandinstruction.

Computer Science Prepared Graduate Statements

PG: 1. Develop, utilize, and evaluate algorithms, to model and solve problems.

Agree: (no comment)

Agree: (no comment)

Agree: (no comment)

Disagree: (no comment)

Agree: (no comment)

Disagree: (no comment)

Agree: (no comment)

Agree: (no comment)

Agree: (no comment)

PG: 2. Systematically analyze a problem using decomposition and abstraction to formulate a solution.

Agree: (no comment)

Agree: Add "pattern recognition" to list in addition to abstraction and decomposition

Disagree: (no comment)

Agree: (no comment)

Disagree: (no comment)

Agree: (no comment)

Agree: (no comment)

Agree: (no comment)

PG: 3. Represent, analyze and visualize data in order to generate new knowledge and capability.

Agree: (no comment)

Disagree: use the Oxford comma

Agree: (no comment)

Agree: (no comment)

Disagree: (no comment)

Comment: Visualize data is a bit abstract, and I'd like to see more clarity for this. i.e. represent data visually.

Agree: (no comment)

Agree: (no comment)

PG: 4. Use systems thinking to describe networks and common software and hardware components.

Agree: (no comment)

Neutral: (no comment)

Agree: (no comment)

Neutral: (no comment)

Disagree: (no comment)

Agree: (no comment)

Agree: (no comment)

Agree: (no comment)

PG: 5. Develop systems solutions from a set of specifications to complete a design process.

Agree: (no comment)

Comment: "engineering" design process?

Agree: (no comment)

Agree: (no comment)

Disagree: (no comment)

Agree: (no comment)

Agree: (no comment)

Agree: (no comment)

PG: 6. Recognize and apply security methodologies to ensure the prevention of exploitation, data protection, and recovery of computing systems following interruption of service.

Agree: (no comment)

Comment: This could be a challenge to demonstrate given current restrictions on student network access at schools.

Neutral: I believe that this might be a little more advanced for most students in High School.

Agree: This is very important and not given enough attention in contemporary learning. Please do not let anyone dissuade you from keeping and emphasizing this learning goal.

Neutral: (no comment)

Disagree: (no comment)

Agree: (no comment)

Agree: (no comment)

Agree: (no comment)

PG: 7. Design and create programs, individually and collaboratively, for a variety of disciplines.

Neutral: (no comment)

Agree: "programs and applications"

Neutral: Creating programs requires a bit more complexity that e do not currently cover in High School.

Agree: (no comment)

Disagree: (no comment)

Comment: More specificity with regard to programs to be created might be helpful. i.e. digital programs

Agree: (no comment)

Agree: (no comment)

PG: 8. Create computational artifacts that consider security from tampering, malicious or otherwise.

Agree: (no comment)

Neutral: (no comment)

Agree: (no comment)

Disagree: (no comment)

Disagree: (no comment)

Agree: (no comment)

Neutral: Why not have additional standards for - Testing and Refining Computational Artifacts Since there is not a general place to write comments, - Communicating about Computing - needs to be a category!

Agree: (no comment)

Computer Science High School

Standard: 1. Computational Thinking

GLE: 1. Computational thinking is used to create algorithmic solutions to real world problems.

Disagree: Test 2

Agree: (no comment)

(PG Feedback) Neutral: Test 1

(PG Feedback) Agree:

(PG Feedback) Neutral: These comments are general about Computer Science standards - there needs to be middle school and elementary level standards too, standards seemed light on some topic areas (quality assurance/testing, software product management, using system to manage process such as Agile, usability), CSTA standards have more coverage of grade levels and broader topic areas and could be used as guide - http://www.csteachers.org/page/standards

(PG Feedback) Comment: We strongly support the proposed high school Computer Science standards and are appreciate of the board and committee's work to ensure high school students are well-prepared to meet the demands of post-secondary opportunities. However, we recommend that the board also develop K-8 standards for Colorado students that will prepare them for these rigorous standards in high school. Without a clear pathway to meeting the foundational knowledge required to meet the high school standards, our concern is that students will not be prepared (and teachers unable to prepare) for meeting these standards once they are in high school. We welcome the opportunity to discuss this with the board or any

other stakeholders moving forward, and thank you for the opportunity to provide feedback on these standards as written.

(PG Feedback) Agree: real-world computing problems

Evidence Outcome: Identify and create different types of algorithms (sort, search, etc).

Comment: first

Neutral: less specific or include logic, loops, recursion, etc A little misleading.

Evidence Outcome: Predict the outcome of different types of algorithms.

Neutral: Describe and predict....

Evidence Outcome: Create or adapt algorithms to solve problems for multiple purposes (i.e., personal interests, client needs).

Neutral: Create, adapt, test

Evidence Outcome: Use an algorithm that involves mathematical operations and functions to solve problems.

Comment: test 5

Neutral: can you include non-mathematical please Not ALL of CS is mathematical, and lots of us are Math folks!

Evidence Outcome: Use an iterative approach to utilizing and/or developing an algorithm.

Comment: 555

Neutral: developing, testing, and utilizing! Not much adherence in these standards to the SW life cycle, etc. Very, very useful to give kids this framework and look to this as being part of their work from school to career!

Evidence Outcome: Recognize problems that cannot be solved computationally.

Neutral: Is this really a primary standard?

Evidence Outcome: Identify and describe algorithms that exist within their personal lives.

Disagree: a, b & f, g use low-level depth of knowledge. What is the measurable skill?

Neutral: . and that are studied personal lives? should that be a primary description? sort of odd, and certainly does not make the standards adaptable to CTE

Comment: last

Comment: g and h are the same

Disagree: Duplicate of g?

Comment: repeat of g.

Neutral: . and that are studied personal lives? should that be a primary description? sort of odd, and certainly does not make the standards adaptable to CTE

Computer Science Practices:

Comment: Please consider including more language regarding ethics from the k12cs.org framework. "Culture The design and use of computing technologies and artifacts can improve, worsen, or maintain inequitable access to information and opportunities. Social Interactions Many aspects of society, especially careers, have been affected by the degree of communication afforded by computing. The increased connectivity between people in different cultures and in different career fields has changed the nature and content of many careers. Safety, Law, and Ethics Laws govern many aspects of computing, such as privacy, data, property, information, and identity. These laws can have beneficial and harmful effects, such as expediting or delaying advancements in computing and protecting or infringing upon people's rights. International differences in laws and ethics have implications for computing.

SUBHEADING NOT FOUND

Neutral: Test 5

Colorado Essential Skills:

Neutral: test

GLE: 2. Algorithms can be represented and used in different ways (e.g., languages, diagrams, pseudocode).

Comment: Understanding how algorithms can be used in different settings would seem better to engage with before creating algorithms. 1. Computational thinking is used to create algorithmic solutions to real world problems. This might allow students the opportunity to think of a variety of applications prior to creating an algorithm.

Agree: (no comment)

(PG Feedback) Agree:

Evidence Outcome: Recognize that different algorithms can be used to solve the same problem.

Disagree: Again, "Recognize" is not a measurable skill. "Recognize that different algorithms can be used to solve the same problem." could be rewritten as "choose multiple applicable solutions for algorithms".

Evidence Outcome: Use multiple methods to represent an algorithm (e.g., diagram, programming language, unplugged).

Agree: (no comment)

Elaboration on the GLE:

Comment: Test

GLE: 3. Algorithm development and use is an ongoing process that involves adapting, critiquing, and troubleshooting programs and/or processes.

Agree: (no comment)

Agree: (no comment)

Evidence Outcome: Describe pros and cons of the performance of algorithms for the same task.

Agree: (no comment)

Evidence Outcome: Use an iterative approach to developing an algorithm.

Agree: (no comment)

Evidence Outcome: Test and troubleshoot so that algorithms produce reasonable results.

Agree: (no comment)

GLE: 4. Large, complex problems can be broken down into smaller, more manageable components.

Agree: (no comment)

Evidence Outcome: Explain how the process of decomposition is iterative and used to solve problems.

Neutral: change "explain" to "demonstrate"

GLE: 5. Abstraction is used to reduce complexity of larger problems by focusing on main ideas.

Agree: and identifying smaller problems

GLE: 6. Data can be represented in different ways for storage and exchange.

Agree: (no comment)

Evidence Outcome: Identify different types of data that are exchanged and produced by computers (e.g., protocols).

Neutral: "identify and use"

Evidence Outcome: Discuss various data structures/techniques for storing and processing data (i.e., arrays, lists, tables).

Disagree: Instead of "discuss" use "compare and contrast" or "evaluate"

GLE: 7. Many problems appropriate for solving with a computer are organized around patterns.

Agree: (no comment)

(PG Feedback) Comment: AP Computer Science Big Idea #1 is Creativity. STEAM, or Science, Technology, Engineering, Art, and Mathematics is gaining credibility and global acceptance as an educational paradigm. Please include the goal for fostering student creativity in the CS standards, perhaps in this section. Thank you.

Evidence Outcome: Analyze computer programs to identify patterns within the program.

Disagree: This student outcome is vague and not measurable.

Evidence Outcome: Provide multiple versions of data visualization in order to deepen problem analysis.

Agree: (no comment)

GLE: 8. Data from a computer program can be visually presented to better understand and articulate solutions to a problem.

Agree: (no comment)

Evidence Outcome: Design visualizations using the appropriate tool(s) with the end user in mind.

Agree: (no comment)

Evidence Outcome: Provide multiple versions of data visualization in order to deepen problem analysis.

Agree: (no comment)

Standard: 2. Computing Systems and Networks

GLE: 1. Communication between computers (and over the Internet) can be configured in many different ways and consist of several hardware and software components.

Agree: (no comment)

(PG Feedback) Comment: Does systems thinking include teaching sustainability practices and awareness in computer science? It might be useful to include sustainability somewhere in this standard.

(PG Feedback) Agree:

GLE: 2. Computer hardware, the lowest level of a computer system, consists of many different parts, each providing a specialized function.

Agree: (no comment)

(PG Feedback) Agree:

Evidence Outcome: Explain the difference between memory and disc storage, internal and external storage, i.e., Random Access Memory (RAM), flash, cloud.

Disagree: This seems low level for high school. Would this be more appropriate at the middle school level?

Evidence Outcome: List and explain the common working parts of a computer.

Disagree: This seems low level for high school. Would this be more appropriate at the middle school level, or perhaps even upper elementary?

GLE: 3. Computer software is written for specific purposes.

Agree: (no comment)

(PG Feedback) Agree:

Evidence Outcome: Explain what an operating system is, and why it is important for a computer or computing device (e.g., Linux, Windows, iOS).

Disagree: duplicate of b.

Comment: Duplicate of b.?

GLE: 4. Systems thinking is a way of holistically examining the various components and use cases that go into a given design.

Agree: (no comment)

(PG Feedback) Agree:

GLE: 5. Client considerations drive system design.

Agree: (no comment)

(PG Feedback) Agree:

GLE: 6. Robust computing systems require multiple methods of recovery.

Agree: (no comment)

(PG Feedback) Comment: Wording is a bit awkward on this one. Perhaps: Recognize and apply security methodologies to prevent exploitation, protect data, and recover computing systems following interruption of service.

(PG Feedback) Agree:

GLE: 7. Robust computing systems require data protection.

Agree: (no comment)

(PG Feedback) Agree:

Standard: 3. Computer Programming

GLE: 1. The creation of a computer program requires a design process.

Agree: (no comment)

(PG Feedback) Agree:

(PG Feedback) Neutral: No mention in any of the standards about the software life cycle which is a major part of learning CS, programming and design, development, and testing of programs/software This should not be implied, but very specific!

Evidence Outcome: Understand and apply core programming concepts.

Agree: (no comment)

GLE: 2. The process of programming involves identifying and solving computational problems.

Comment: Do we need an evidence outcome around "identifying" computational problems?

Agree: (no comment)

Evidence Outcome: Write code per selected design.

Neutral: Write computer program based on design using a selected language Please be consistent and use programming, not coding. Coding - a new term, but the CS university programs and professional world andindustry still refer to programming and computer programs

Evidence Outcome: Create code comments to communicate to other developers and ensure documentation of code.

Agree: I agree to this revision because not having comments in your code is bad practice, especially if your program is open-source. Teaching about comments is incredibly useful and should be needed.

Neutral: Computer program comments

Evidence Outcome: Use various troubleshooting and debugging techniques to improve code.

Agree: improve computer program

Evidence Outcome: Create appropriate variables to store and retrieve data.

Agree: ... in the computer program Always be specific, otherwise this will be subject to interpretation in the future

Elaboration on the GLE:

Comment: Is this elaboration a better fit for the design process standard preceding this one?

GLE: 3. Collaborative tools, methods, and strategies can be used to design, develop, and update computational artifacts.

Agree: design, develop, TEST, and update Please do not diminish the very important part of programming - testing!

Evidence Outcome: Determine when to integrate collaborative strategies to improve programming outputs.

Agree: Collaboration in programming is a wonderful idea to integrate! Great minds think alike.

Neutral: Have no idea what this is

GLE: 4. Client-based design requirements and feedback are essential to a quality computational product or service.

Agree: Like this, but as math teachers are recruited with no CTE or industry experience, this may require a lot of PD for teachers

Computer Science Practices:

Neutral: develop and use abstractions does not fit here, this is more a part of programming and design not client feedback etc. In practice, this would never happen

GLE: 5. Computing solutions can have impacts (personal, ethical, social, economic, and cultural) based on their use.

Agree: This one is very important, mainly depending on the product. If you don't clean up your code you could end up with exploits, which could mean less safety. Teaching about security measures is huge!

Agree: (no comment)

GLE: 6. Security and software licensing can present constraints and restrictions in computational design and development.

Agree: (no comment)