*Please note: These are screen shots of the October-December, 2017 online feedback system window for reference only. These are not the final proposed revisions.
# Table of Contents

**Grade Level: High School**

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Prepared Graduate Statements</td>
<td>3</td>
</tr>
<tr>
<td>Computational Thinking</td>
<td>4-10</td>
</tr>
<tr>
<td>Computing Systems and Networks</td>
<td>11-17</td>
</tr>
<tr>
<td>Computer Networks</td>
<td>18-22</td>
</tr>
</tbody>
</table>
Review the Prepared Graduate Statements
Computer Science

Instructions
In this section, you have the opportunity to provide feedback on the content of the Prepared Graduate Statements (PGS).

On the next page, as you review the grade level expectations, you will be able to provide feedback on the alignment of the PGS with the given grade level expectation.

To leave feedback, click on the comment icon next to any item. You can then offer feedback and comments.

Once you save your feedback, the icon will change color and show as a checkbox so you can keep track of your progress.

About Prepared Graduate Statements (PGS)
All of Colorado’s Academic Standards were designed “backwards” from Prepared Graduate Statements. These statements were formerly known as Prepared Graduate Competencies but have been changed to reduce confusion with competency-based learning systems of instruction and assessment practices. The PGS identify the concepts and skills that all students who complete the Colorado education system must master to ensure their success in a postsecondary and workforce setting for each content area.

Each grade level expectation of the Colorado Academic Standards aligns to one or more of the PGS.

| 1. Develop, utilize, and evaluate algorithms, to model and solve problems. |
| 2. Systematically analyze a problem using decomposition and abstraction to formulate a solution. |
| 3. Represent, analyze and visualize data in order to generate new knowledge and capability. |
| 4. Use systems thinking to describe networks and common software and hardware components. |
| 5. Develop systems solutions from a set of specifications to complete a design process. |
| 6. Recognize and apply security methodologies to ensure the prevention of exploitation, data protection, and recovery of computing systems following interruption of service. |
| 7. Design and create programs, individually and collaboratively, for a variety of disciplines. |
| 8. Create computational artifacts that consider security from tampering, malicious or otherwise. |
Computer Science

Grade Level: High School
Standard: 1. Computational Thinking

Instructions
To leave feedback, click on the comment icon next to any item. You can then offer feedback and comments. Once you save your feedback, the icon will change color and show as a checkbox so you can keep track of your progress.

Prepared Graduates
1. Develop, utilize, and evaluate algorithms to model and solve problems.

Grade Level Expectation: High School
1. Computational thinking is used to create algorithmic solutions to real-world problems.

Evidence Outcomes

Students Can:

a. Identify and create different types of algorithms (sort, search, etc.)

b. Predict the outcomes of different types of algorithms.

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

d. Use an algorithm that involves mathematical operations and functions to solve problems.

e. Use an iterative approach to utilizing and/or developing an algorithm.

f. Recognize problems that cannot be solved computationally.

g. Identify and describe algorithms that exist within their personal lives.

h. Identify and describe algorithms that exist within their personal lives.

Academic Context and Connections

Colorado Essential Skills:
1. Entrepreneurial Skills: Critical thinking/problem solving, Creativity/Innovation, Inquiry and analysis


3. Civic/Interpersonal Skills: Communication; Global/cultural awareness

4. Professional Skills: Information literacy; Use of information/communications technologies; Perseverance/resilience; Productivity/accountability; Leadership

Elaboration on the GLE:
1. Central to computational thinking are the processes of generalization and decomposition, with an eye toward the technology that will be used to solve the problem. This planning and abstraction process should also include students decomposing complex problems into manageable subproblems that could potentially be solved with programs or procedures that already exist. As students develop algorithms they should identify procedures and/or functions that are used multiple times within a program to repeat groups of instructions (CS 3A-AP-17 & Z-AP-14).

Computer Science Practices:
1. Fostering an Inclusive Computing Culture
2. Recognizing and Defining Computational Problems
3. Creating Computational Artifacts
## Prepared Graduates

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

## Grade Level Expectation: High School

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

## Evidence Outcomes

### Students Can:

- a. Recognize that different algorithms can be used to solve the same problem.
- b. Illustrate the flow of execution of an iterative algorithm (e.g., recursion).
- c. Explain the value of heuristic algorithms to model ways to solve problems.
- d. Adapt algorithms used in one problem to solve a related or different problem.
- e. Use multiple methods to represent an algorithm (e.g., diagram, programming language, unplugged).

## Academic Context and Connections

### Colorado Essential Skills:

1. Entrepreneurial Skills: Critical thinking/problem solving; Creativity/Innovation; Inquiry and analysis
2. Personal Skills: Self-awareness; Initiative/Self Direction
3. Civic/Interpersonal Skills: Communication; Global/cultural awareness
4. Professional Skills: Information literacy; Use information/communications technologies; Perseverance/resilience; Productivity/accountability; Leadership

### Elaboration on the GLE:

1. Students should use pseudocode, diagrams and/or flowcharts to organize and sequence an algorithm that addresses a problem. Representing algorithms in alternative forms supports the planning phase of the design process and helps students see various ways to structure an algorithm (CSTA 2-AP-10).

### Computer Science Practices:

1. Recognizing and defining computational problems
2. Communicating about computing
Prepared Graduates

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

Grade Level Expectation: High School

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

Evidence Outcomes

Students Can:

a. Describe pros and cons of the performance of algorithms for the same task.

b. Use an iterative approach to developing an algorithm.

c. Test and troubleshoot so that algorithms produce reasonable results.

Academic Context and Connections

Colorado Essential Skills:

1. Entrepreneurial Skills: Critical thinking/problem solving; Creativity/Innovation; Inquiry and analysis

2. Personal Skills: Self-awareness; Initiative / Self Direction

3. Civic/Interpersonal Skills: Communication; Global/cultural awareness

4. Professional Skills: Information literacy; Use information/communications technologies; Perseverance/resilience; Productivity/accountability; Leadership

Elaboration on the GLE:

1. Testing and refinement is the deliberate and iterative process of improving a computational artifact. This process includes debugging (identifying and fixing errors) and comparing actual outcomes to intended outcomes. Students should respond to the changing needs and expectations of end users and improve the performance, reliability, usability, and accessibility of artifacts. For example, students could incorporate feedback from a variety of end users to help guide the size and placement of menus and buttons in a user interface (CSTA 3A-AP-21).

Computer Science Practices:

1. Testing and refining computational artifacts

2. Creating computational artifacts

3. Recognizing and defining computational problems
Prepared Graduates
2. Systematically analyze a problem using decomposition and abstraction to formulate a solution.

Grade Level Expectation: High School
4. Large, complex problems can be broken down into smaller, more manageable components.

Evidence Outcomes
Students Can:

a. Explain how the process of decomposition is iterative and used to solve problems.

b. Formulate possible solutions based on the decomposition of a problem.

Academic Context and Connections
Colorado Essential Skills:
1. Entrepreneurial Skills: Critical thinking/problem solving; Creativity/Innovation; Inquiry and analysis
2. Personal Skills: Self-awareness; Initiative / Self Direction
3. Civic/Interpersonal Skills: Communication; Global/cultural awareness
4. Professional Skills: Information literacy; Use information/communications technologies; Perseverance/resilience; Productivity/accountability; Leadership

Elaboration on the GLE:
1. At this level, students should decompose complex problems into manageable subproblems that could potentially be solved with programs or procedures that already exist. For example, students could create an app to solve a community problem by connecting to an online database through an application programming interface (API) (CSTA 3A-AP-17).

Computer Science Practices:
1. Communicating about computing
2. Recognizing and defining computational problems
Prepared Graduates
2. Systematically analyze a problem using decomposition and abstraction to formulate a solution.

Grade Level Expectation: High School
5. Abstraction is used to reduce complexity of larger problems by focusing on main ideas.

Evidence Outcomes
Students Can:
a. Describe how abstraction is central to computational thinking.
b. Identify and prioritize the most relevant parts of a problem while filtering out extraneous details.
c. Demonstrate different ways to represent key problem components.

Academic Context and Connections
Colorado Essential Skills:
1. Entrepreneurial Skills: Critical thinking/problem solving; Creativity/Innovation; Inquiry and analysis
2. Personal Skills: Self-awareness; Initiative / Self Direction
3. Civic/Interpersonal Skills: Communication; Global/cultural awareness
4. Professional Skills: Information literacy; Use information/communications technologies; Perseverance/resilience; Productivity/accountability; Leadership

Elaboration on the GLE:
1. Abstraction is a necessary part of modeling, problem solving, and computational thinking; it requires the identification of key aspects of a given context to formulate and solve a problem of interest. Students might select an embedded device such as a car stereo, identify the types of data (radio station presets, volume level) and procedures (increase volume, store/recall saved station, mute) it includes, and explain how the implementation details are hidden from the user (CSTA 3A-CS-01).

Computer Science Practices:
1. Developing and using abstractions
2. Communicating about computing
3. Testing and refining computational artifacts
Prepared Graduates

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

Grade Level Expectation: High School

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

Evidence Outcomes

Students Can:

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

b. Represent data using multiple encoding schemes (e.g., RGB, Hex, HSB, ASCII, Unicode).

c. Evaluate the trade-offs for how data elements are organized and where data are stored (e.g., PNG/GIF, structured/unstructured).

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

Academic Context and Connections

Colorado Essential Skills:

1. Entrepreneurial Skills: Critical thinking/problem solving; Creativity/Innovation; Inquiry and analysis

2. Personal Skills: Self-awareness; Initiative/Self Direction

3. Civic/Interpersonal Skills: Communication; Global/cultural awareness

4. Professional Skills: Information literacy; Use information/communications technologies; Perseverance/resilience; Productivity/accountability; Leadership

Elaboration on the GLE:

1. People make choices about how data elements are organized and where data is stored (e.g., convert hexadecimal color codes to decimal percentages, ASCII/Unicode representation, and logic gates (CSTA 3A-DA-09)). These choices affect cost, speed, reliability, accessibility, privacy, and integrity. Students should evaluate whether a chosen solution is most appropriate for a particular problem. Students might consider the cost, speed, reliability, accessibility, privacy, and integrity tradeoffs between storing photo data on a mobile device versus in the cloud (CSTA 3A-DA-10).

Computer Science Practices:

1. Communicating about computing

2. Creating computational artifacts

3. Testing and refining computational artifacts
Prepared Graduates

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

Grade Level Expectation: High School

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

Evidence Outcomes

Students Can:

a. Analyze computer programs to identify patterns within the program.

b. Provide multiple versions of data visualization in order to deepen problem analysis.

c. Interpret and analyze data to make informed decisions.

Academic Context and Connections

Colorado Essential Skills:

1. Entrepreneurial Skills: Critical thinking/problem solving; Creativity/Innovation; Inquiry and analysis

2. Personal Skills: Self-awareness; Initiative/Self Direction

3. Civic/Interpersonal Skills: Communication; Global/cultural awareness

4. Professional Skills: Information literacy; Use information/communications technologies; Perseverance/resilience; Productivity/accountability; Leadership

Elaboration on the GLE:

1. One of the most powerful features of computational thinking is using technological tools to make sense of natural and social phenomena. Coding and analytic techniques can be used to identify and visualize patterns in complex data. For example, students could be asked to identify trends in a data set representing social media interactions, movie reviews, or shopping patterns (CSTA 3B-DA-05).

Computer Science Practices:

1. Recognizing and defining computational problems

2. Testing and refining computational artifacts
Prepared Graduates

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

Grade Level Expectation: High School

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

Evidence Outcomes

Students Can:

a. Analyze computer output in different forms (e.g., plain text, CSV, graphs, images).

b. Design visualizations using the appropriate tool(s) with the end user in mind.

c. Provide multiple versions of data visualization in order to deepen problem analysis.

Academic Context and Connections

Colorado Essential Skills:

1. Entrepreneurial Skills: Critical thinking/problem solving; Creativity/Innovation; Inquiry and analysis; Risk taking

2. Personal Skills: Initiative/Self Direction

3. Civic/Interpersonal Skills: Global/cultural awareness

4. Professional Skills: Information literacy; Use information/communications technologies

Elaboration on the GLE:

1. People transform, generalize, simplify, and present large data sets in different ways to influence how other people interpret and understand the underlying information. Examples include visualization, aggregation, rearrangement, and application of mathematical operations. People use software tools or programming to create powerful, interactive data visualizations and perform a range of mathematical operations to transform and analyze data. Students should model phenomena as systems, with rules governing the interactions within the system and evaluate these models against real-world observations. For example, flocking behaviors, queueing, or life cycles (CSTA 3A-DA-11).

Computer Science Practices:

1. Recognizing and defining computation problems

2. Creating computational artifacts

3. Fostering an inclusive computing culture
Computer Science
Grade Level: High School
Standard: 2. Computing Systems and Networks

Instructions
To leave feedback, click on the comment icon ( ) next to any item. You can then offer feedback and comments. Once you save your feedback, the icon will change color and show as a checkbox ( ) so you can keep track of your progress.

Prepared Graduates
4. Use systems thinking to describe networks and common software and hardware components.

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

Evidence Outcomes
Students Can:
- Describe key protocols and underlying processes of Internet-based services, (i.e., http) and discuss impact of technology change on communication protocols.
- Illustrate and describe the basic components and various network types and topologies (i.e., personal, local, metropolitan, and wide).
- Explain the difference between decimal, hexadecimal, octal, and binary number formats and how they are used in computing environments.

Colorado Essential Skills:
1. Entrepreneurial Skills: Critical Thinking/Problem Solving; Inquiry Analysis; Risk Taking
2. Personal Skills: Adaptability/Flexibility; Perseverance & Adaptability
3. Civic/Interpersonal Skills: Communication
4. Professional Skills: Information Literacy; Use Information and Communications Technologies

Elaboration on the GLE:
1. Computing is at its most powerful when devices are connected via a network. Networks are comprised of various hardware and software components which have specific functions within the network. For example, individual devices are assigned an address that uniquely identifies it on the network; routers function by comparing IP addresses to determine the pathways packets should take to reach their destination; and switches function by comparing MAC addresses to determine which computers or network segments will receive frames (CSTA, 3A-NI-04). Each device is assigned an address that uniquely identifies it on the network. Routers function by comparing IP addresses to determine the pathways packets should take to reach their destination. Switches function by comparing MAC addresses to determine which computers or network segments will receive frames. Students could use online network simulators to experiment with these factors (CSTA, 3A-NI-04).

Computer Science Practices:
1. Developing and Using Abstractions
Prepared Graduates

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

Grade Level Expectation: High School

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

Evidence Outcomes

Students Can:

a. Explain the difference between memory and disk storage, internal and external storage, i.e., Random Access Memory (RAM), flash, cloud.

b. List and explain the common working parts of a computer.

c. Explain how to maintain safety when working on PCs, i.e., electromagnetic precautions.

d. Describe how computing devices are engineered for fault tolerance and reliability, and identify potential sources of weakness (e.g., redundant power supplies, RAID, SAN/NAS connections).

Academic Context and Connections

Colorado Essential Skills:

1. Entrepreneurial Skills: Critical Thinking/Problem Solving; Inquiry Analysis; Risk Taking

2. Personal Skills: Adaptability/Flexibility; Perseverance & Adaptability

3. Civic/Interpersonal Skills: Communication

4. Professional Skills: Information Literacy; Use Information and Communications Technologies

Elaboration on the GLE:

1. At its most basic level, a computer is composed of physical hardware and electrical impulses. A computing system is composed of components such as the central processor (executes commands), memory (for temporary storage of data), hard disk (stores data), mainboard (provides communication between components and peripherals) and network interface (communicates with other devices) and power supply (CSTA 3A-CS-02).

Computer Science Practices:

1. Developing and Using Abstractions
Prepared Graduates
4. Use systems thinking to describe networks and common software and hardware components.

Grade Level Expectation: High School
3. Computer software is written for specific purposes.

Evidence Outcomes

Students Can:

a. Identify and differentiate between different kinds of software (i.e., operating systems vs. applications) and their purposes.

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

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

d. Describe how software interacts with hardware to complete tasks.

Academic Context and Connections

Colorado Essential Skills:
1. Entrepreneurial Skills: Critical Thinking/Problem Solving; Inquiry Analysis; Risk Taking
2. Personal Skills: Adaptability/Flexibility; Perseverance & Adaptability
3. Civic/Interpersonal Skills: Communication
4. Professional Skills: Information Literacy; Use Information and Communications Technologies

Elaboration on the GLE:
1. System software manages a computing device’s resources (CSTA 3A-CS-02). Students should recognize that there is a variety of software user interfaces, and that different software exist for different purposes (i.e., operating system vs. application).

Computer Science Practices:
1. Communicating About Computing
Prepared Graduates

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

Grade Level Expectation: High School

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

Evidence Outcomes

Students Can:

a. Explain the integration of hardware, software and network communications components to create a networked system.

b. Summarize security approaches using a systems approach perspective.

Academic Context and Connections

Colorado Essential Skills:

1. Entrepreneurial Skills: Critical Thinking/Problem Solving; Inquiry Analysis; Risk Taking

2. Personal Skills: Adaptability/Flexibility; Perseverance & Adaptability

3. Civic/Interpersonal Skills: Communication

4. Professional Skills: Information Literacy; Use Information and Communications Technologies

Elaboration on the GLE:

1. By itself, a computer is just a dumb piece of hardware. It is not until an operating system is loaded on it that the computer becomes useful. The OS handles the operation of the hardware in conjunction with the software applications a user has loaded. System software manages a computing device’s resources so that software can interact with hardware. (CSTA 3A-CS-02) Systems thinking utilizes concepts and tools that helps people to understand the makeup of large systems, like computer networks, to meet user needs/requirements, and to make sure computer systems are secure.

Computer Science Practices:

1. Developing and using abstractions.

2. Recognizing and defining computational problems

3. Communicating about computing
Prepared Graduates

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

Grade Level Expectation: High School

5. Client considerations drive system design.

Evidence Outcomes

Students Can:

a. Identify client’s problems/needs.

b. Articulate design requirements back to client.

c. Illustrate options for considerations and develop conceptual model.

d. Perform system analysis based on client considerations.

Academic Context and Connections

Colorado Essential Skills:

1. Entrepreneurial Skills: Critical Thinking/Problem Solving; Inquiry Analysis; Risk Taking

2. Personal Skills: Adaptability/Flexibility; Perseverance & Adaptability; Initiative and Self-Direction

3. Civic/Interpersonal Skills: Communication

4. Professional Skills: Information Literacy; Use Information and Communications Technologies

Elaboration on the GLE:

1. Software engineers plan and develop programs for broad audiences using a software life cycle process (CSTA 3B-AP-17). Similarly, systems architects use, plan, and develop networks to meet specific client needs.

Computer Science Practices:

1. Recognizing and Defining Computational Problems

2. Communicating about computing

3. Creating computational artifacts

4. Testing and refining computational artifacts
Prepared Graduates

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

Grade Level Expectation: High School

6. Robust computing systems require multiple methods of recovery.

Evidence Outcomes

Students Can:

a. Identify different ways that systems might lose data or functionality.

b. Describe elements of an effective backup system.

c. Compare backup systems for computer users, or a network.

d. List the various backup methodologies (e.g., full, differential), and why one would pick one over the other, or use all.

e. Explain the ways an organization would continue to operate in light of a systems failure.

Academic Context and Connections

Colorado Essential Skills:

1. Entrepreneurial Skills: Critical Thinking/Problem Solving; Inquiry Analysis; Risk Taking

2. Personal Skills: Adaptability/Flexibility; Perseverance & Adaptability

3. Civic/Interpersonal Skills: Communication

4. Professional Skills: Information Literacy; Use Information and Communications Technologies

Elaboration on the GLE:

1. The timely and reliable access to data and information services by authorized users, referred to as availability, is ensured through adequate bandwidth, backups, and other measures (CSTA 3A-NI-06). Students should understand that an “interruption of service” can come about through disasters, hacking and other deliberate exploitations, power issues, and other identifiable problems (e.g., hurricanes). The process of identifying interruptions in service is an important skill for those wanting to work in Information Technology (IT). Backing up a system means that you denote a process in which your computer copies certain data to another safe spot (e.g., another drive, the cloud). Backups are also used in Information Technology (IT) shops in various companies, governmental agencies, and educational institutions.

Computer Science Practices:

1. Recognizing and Defining Computational Problems

2. Communicating about computing

3. Collaborating about computing
**Prepared Graduates**
6. Recognize and apply security methodologies to ensure the prevention of exploitation, data protection, and recovery of computing systems following interruption of service.

**Grade Level Expectation: High School**
7. Robust computing systems require data protection.

**Evidence Outcomes**
- a. Identify examples of threats to systems and data.
- b. Describe the process by which intruders gain entry into a production system (e.g., reconnaissance).
- c. Describe and compare methods to test/validate how well systems and data are protected.
- d. Investigate different career pathways relating to systems security.

**Academic Context and Connections**

**Colorado Essential Skills:**
1. Entrepreneurial Skills: Critical Thinking/Problem Solving; Inquiry Analysis; Risk Taking
2. Personal Skills: Adaptability/Flexibility; Perseverance & Adaptability
3. Civic/Interpersonal Skills: Communication; Character; Global/Cultural Awareness
4. Professional Skills: Information Literacy; Use Information and Communications Technologies; Career Awareness

**Elaboration on the GLE:**
1. Security measures may include physical security tokens, two-factor authentication, and biometric verification. Potential security problems, such as denial-of-service attacks, ransomware, viruses, worms, spyware, and phishing, exemplify why sensitive data should be securely stored and transmitted. Students should systematically evaluate the feasibility of using computational tools to solve given problems or subproblems, such as long, complex passwords (CSTA 3A-NI-06). See also CSTA 3B-NI-04 and 3B-AP-10).

**Computer Science Practices:**
1. Recognizing and Defining Computational Problems
2. Communicating about computing
3. Collaborating about computing
1. The creation of a computer program requires a design process.

Evidence Outcomes

Students Can:

a. Analyze and apply a design methodology to identify constraints and requirements of an identified problem.

b. Utilize tools and resources such as pseudocode, flowcharts, wireframes, etc. as part of the design process.

c. Determine and use graphical or text-based languages.

d. Understand and apply core programming concepts.

Academic Context and Connections

Colorado Essential Skills:

1. Entrepreneurial Skills: Critical thinking/problem solving; Inquiry/analysis
2. Personal Skills: Adaptability/flexibility
3. Civic/Interpersonal Skills: Communication
4. Professional Skills: Task/time management; Productivity/accountability; Use information/communications technologies

Elaboration on the GLE:

1. Computer programming requires selection of a design methodology (i.e., engineering, software, human-centered) to identify user needs and requirements. Methodologies provide tools for making important design decisions and help programmers manage the iterative process of software design (CSTA 3A-AHP-13).

Computer Science Practices:

1. Recognizing and defining computational problems
2. Developing and using abstractions
3. Creating computational artifacts
4. Communicating about computing

Prepared Graduates

7. Design and create programs, individually and collaboratively, for a variety of disciplines.
Prepared Graduates
7. Design and create programs, individually and collaboratively, for a variety of disciplines.

Grade Level Expectation: High School
2. The process of programming involves identifying and solving computational problems.

Evidence Outcomes
Students Can:
- a. Write code per selected design.
- b. Create code comments to communicate to other developers and ensure documentation of code.
- c. Use various troubleshooting and debugging techniques to improve code.
- d. Create appropriate variables to store and retrieve data.

Academic Context and Connections
Colorado Essential Skills:
1. Entrepreneurial Skills: Creativity/innovation; Inquiry/analysis
2. Personal Skills: Adaptability/flexibility; Perseverance/resilience
3. Civic/Interpersonal Skills: Communication
4. Professional Skills: Task/time management; Productivity/accountability; Use information/communications technologies

Elaboration on the GLE:
1. Using an iterative process helps students to create a high-quality product that meets a client’s design requirements. It guides them through the following cycle: prototyping, testing, feedback from client, revision. Software design is a universal approach that can be used irrespective of programming tools (such as a specific language). Effective design utilizes practices such as commenting to record rationale for specific design decisions (CSTA 3A-AP-21).

Computer Science Practices:
1. Developing and using abstractions
2. Creating computational artifacts
3. Testing and refining computational artifacts
4. Communicating about computing
Prepared Graduates
7. Design and create programs, individually and collaboratively, for a variety of disciplines.

Grade Level Expectation: High School
3. Collaborative tools, methods, and strategies can be used to design, develop, and update computational artifacts.

Evidence Outcomes
Students Can:
a. Determine when to integrate collaborative strategies to improve programming outputs.
b. Identify and analyze a variety of collaborative tools (i.e., commenting, development repositories) in order to determine the appropriateness for intended use.
c. Identify strategies such as peer reviews to test and refine artifacts in development.
d. Determine when to use standard software tools like APIs, libraries, version control repositories, etc.

Academic Context and Connections
Colorado Essential Skills:
1. Entrepreneurial Skills: Creativity/innovation; Inquiry/analysis
2. Personal Skills: Adaptability/flexibility; Perseverance/resilience
3. Civic/Interpersonal Skills: Collaboration/teamwork; Communication; Global / cultural awareness
4. Professional Skills: Task/time management; Use information/communications technologies; Productivity/accountability

Elaboration on the GLE:
1. Collaborative strategies such as peer programming and feedback protocols have students optimally revise computational artifacts (e.g., graphical interfaces, program performance, errors) and help foster an inclusive computing culture which produces artifacts that meet the needs of a broad audience (CSTA 3A-AP-22).

Computer Science Practices:
1. Creating computational artifacts
2. Fostering an inclusive computing culture
3. Testing and refining computational artifacts
4. Collaborating around computing
5. Communicating around computing
Prepared Graduates
7. Design and create programs, individually and collaboratively, for a variety of disciplines.

Grade Level Expectation: High School
4. Client-based design requirements and feedback are essential to a quality computational product or service.

Evidence Outcomes

Students Can:

- a. Understand and apply principles of client-based design.
- b. Guide/advise clients on strategies and solutions best suited for their problem (i.e., type of platform).
- c. Construct effective methods for gathering feedback from clients.
- d. Respond to feedback from clients to improve computing solutions.
- e. Create and share product support documentation for potential users.
- f. Articulate lessons learned as a result of the design and creation process.

Academic Context and Connections

Colorado Essential Skills:

- 1. Entrepreneurial Skills: Critical thinking / problem solving; Inquiry / Analysis; Risk taking
- 2. Personal Skills: Initiative / self-direction; Personal responsibility; Adaptability / flexibility
- 3. Civic/Interpersonal Skills: Communication; Global / cultural awareness; Character
- 4. Professional Skills: Task / time management; Career awareness; Use of information and communication technologies; Productivity/accountability; Leadership

Elaboration on the GLE:
1. By allowing students the opportunity to develop programs at the request of a client or identified real-world situation, students are able to have a more authentic learning experience. Students will pursue learning opportunities that are very similar in nature to experiences they will have in a future computer science career. It is important that students follow protocols and frameworks that they would see in the modern workplace to identify problems, develop a programming solution, and bring their artifact to life for review by outside clients (CSTA 3A-AP-19 & 3A-IC-27).

Computer Science Practices:

1. Recognizing and defining computational problems
2. Develop and using abstractions
3. Testing and refining computational artifacts
4. Communicating about computing
5. Collaborating about computing
Prepared Graduates

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

Grade Level Expectation: High School

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

Evidence Outcomes

Students Can:

a. Investigate and understand privacy, security, and protection laws.

b. Articulate the importance of securing personal data information on encrypted storage systems.

c. Identify and analyze current events to ensure the safety, security, and well-being of all potential clients and end users.

Academic Context and Connections

Colorado Essential Skills:

1. Entrepreneurial Skills: Critical thinking/problem solving; Inquiry and analysis; Risk taking

2. Personal Skills: Self-awareness; Adaptability; Initiative; Personal responsibility; Perseverance/resilience

3. Civic/Interpersonal Skills: Collaboration; Communication; Global/cultural awareness; Civic engagement; Character

4. Professional Skills: Career awareness; Information Literacy; Use information/communications technologies; Self-advocacy

Elaboration on the GLE:

1. As students engage in computer programming it is important for them to be highly aware of the many aspects of cyber and information security. Students need to be aware not only of security loopholes that open their programs up to hacking but also to accidental programming errors or choices that can lead to others security issues as well. Students should do their best to be proactive in their programming but be aware they will need to update code and patch as needed when security vulnerabilities arise. Students should understand the importance of keeping their devices and programs up to date through additional updates and patches but that those as well can lead to other problems. Students want to ensure security is included in their feedback cycle for developed solutions (CSTA 3A-N1-08).

Computer Science Practices:

1. Recognizing and defining computational problems

2. Testing and refining computational artifacts

3. Communicating about computing
## Prepared Graduates

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

## Grade Level Expectation: High School

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

## Evidence Outcomes

### Students Can:

- **a.** Describe how software licensing influences program development.
- **b.** Investigate and develop solutions that discourage online software piracy.
- **c.** Explore and integrate security measures such as encryption, authentication, and verification strategies to secure developed computer programs.
- **d.** Research and abide by intellectual property laws & patents.

## Academic Context and Connections

### Colorado Essential Skills:

1. Entrepreneurial Skills: Critical thinking/problem solving; Inquiry and analysis; Risk taking
2. Personal Skills: Self-awareness; Adaptability; Initiative; Personal responsibility; Perseverance/resilience
3. Civic/Interpersonal Skills: Collaboration; Communication; Global/cultural awareness; Civic engagement; Character
4. Professional Skills: Career awareness; Information Literacy; Use information/communications technologies; Self-advocacy

### Elaboration on the GLE:

1. After finishing a computer program, students should consider how they would potentially distribute their product. Whether they determine to sell it at a price on an app store or distribute it for free, a license of some sort is required and a process for which consumers can access the program. Alternatively, students need to be mindful that pirating occurs and should think about ways they can secure their programs to not be unlawfully distributed such as licensing codes, attachment to connected services, distribute software, etc. Students need to be aware of laws and patents that govern protect intellectual property CSTA 3A-AP-20 & 3A-IC-28.

### Computer Science Practices:

1. Recognizing and defining computational problems
2. Testing and refining a computational artifact
3. Communicating about computing