Skill Struck's Alignment to

Utah State Academic Standards

Our computer science curriculum for grades K-12 aligns with **116/116** Utah Academic Standards.

LEGEND

Standards aligned ✓ Standards not currently aligned ○

Kindergarten

Standard	Status (6/6, 100%)
K.CS.1: Select computing devices that perform a variety of tasks accurately and quickly based on user needs and preferences.	✓
K.NI.1: Model and describe how people connect to other people and information through a network.	✓
K.NI.2: Create patterns to communicate a message.	✓
K.DA.1: Identify and describe patterns in data visualizations, such as charts or graphs, to make predictions.	✓
K.AP.1: Model processes by creating and following algorithms to complete tasks.	✓
K.CT.1: Decompose problems into smaller manageable parts to better understand them.	✓

1st Grade

Standard	Status
	(10/10, 100%)
1**.CS.1:** Operate a variety of computing devices that perform tasks accurately and quickly based on user needs and preferences.	✓
1.CS.2: Explore the functions of common hardware and software components of computing systems.	✓
1.DA.1: Collect and present data in various visual formats.	✓
1.DA.2: Identify and describe patterns in data visualizations (unplugged or digital), such as charts or graphs, to make predictions.	✓
1.AP.1: Demonstrate understanding of the way programs store and manipulate data as variables, such as numbers, words, colors, and images.	✓
1.AP.2: Break down (deconstruct) algorithms and list the steps needed to solve a problem into a sequence of tasks and sub-tasks.	✓
1.AP.3: Create programs with sequences (steps) of commands and simple loops (repeated patterns), to express ideas or address a problem.	✓
1.IC.1: Develop and demonstrate the ability to work respectfully and responsibly with others whether communicating face-to-face or digitally.	✓
1.CT.1: Determine the steps needed to solve a problem and develop a sequence of instructions.	✓
1.CT.2: Recognize similarities between new problems and problems that have been solved in the past.	✓

2nd Grade

Standard	Status (12/12, 100%)
2.CS.1: Describe and solve basic hardware and software problems.	✓
2.NI.1: Explain what a password or pass phrase is, why it is used, and be able to create a secure password.	✓
2.DA.1: Demonstrate how to store, copy, search, retrieve, modify and delete information using a computing device, and define the information stored as data.	✓
2.DA.2: Collect and present data in various visual formats.	✓
2.DA.3: Identify and describe patterns in data visualizations to make predictions.	✓
2.AP.1: Deconstruct the steps needed to solve a task into a sequence of instructions.	✓
2.AP.2: Collaboratively develop plans that describe a program's sequence of events, goals, and expected outcomes.	✓
2.AP.3: Properly credit others when using their ideas and creations while developing programs.	✓
2.AP.4: Debug and solve simple problems within an algorithm or program that includes sequences and simple loops.	✓
2.AP.5: Summarize the steps taken and choices made during the iterative process of program development.	✓
2.IC.1: Describe how technology has impacted society over time.	✓
2.IC.2: Describe rationales for keeping login information private, and for logging off devices appropriately.	✓

Skill Struck

3rd Grade

Standard	Status (13/13, 100%)
3.CS.1: Describe and model how computing devices connect to other components to extend their capabilities and form a system.	✓
3.NI.1: Describe physical and digital security measures for protecting personal information.	✓
3.NI.2: Develop personal patterns of behavior to protect information from unauthorized access.	✓
3.DA.1: Organize and present collected data visually to highlight relationships and support a claim.	✓
3.DA.2: Use data to communicate ideas, highlight relationships, and predict outcomes.	✓
3.AP.1: Create programs that include events, sequences, loops, and simple conditionals to express ideas or address a problem.	✓
3.AP.2: Modify a previously created program that uses variables to store and modify data.	✓
3.AP.3: Test and debug a program or algorithm to ensure it accomplishes the intended task.	✓
3.AP.4: Perform different roles when collaborating with peers during the design, implementation, and review stages of program development.	✓
3.AP.5: Use an iterative design process to plan and develop a program by considering the perspectives and preferences of others.	✓
3.AP.6: Create programs by incorporating smaller portions of existing programs to develop something new or add more advanced features.	✓
3.IC.1: Evaluate how computing technologies have changed the world, and express how those technologies influence, and are influenced by, cultural practices.	✓
3.IC.2: Describe reasons creators might limit the use of their work.	✓
3.CT.1: Decompose problems into smaller manageable tasks which may themselves be decomposed.	✓
3.CT.2: Recognize common patterns between problems and recurring patterns within problems.	✓

Skill Struck

4th Grade

Standard	Status
	(11/11, 100%)
4.CS.1: Demonstrate how computer hardware and software work together as a system to accomplish tasks.	✓
4.NI.1: Model how information is broken down into smaller pieces called packets and transmitted through multiple devices over physical or wireless paths and reassembled at the destination.	✓
4.DA.1: Select, organize, and categorize data and represent that data visually to provide clarity or support a claim.	✓
4.DA.2: Use data to highlight and propose relationships, predict outcomes, or communicate ideas.	✓
4.AP.1: Compare and refine multiple algorithms for the same task, using computer and non-computer languages, and determine which is the most appropriate.	✓
4.AP.2: Create programs that include events, loops, and conditionals.	✓
4.AP.3: Decompose problems into smaller, manageable tasks which may be then be broken down further.	✓
4.AP.4: Test and debug a program or algorithm to ensure it accomplishes the intended task.	✓
4.IC.1: Evaluate computing technologies that have changed the world and express how those technologies influence and are influenced by cultural practices.	✓
4.IC.4: Propose ways to improve the accessibility and usability of technology products for the diverse needs and wants of users.	✓
4.CT.1: Determine specific aspects of patterns between or within problems that can be abstracted out to leave only the common or important elements.	✓

Skill Struck

5th Grade

Standard	Status (14/14, 100%)
5.CS.1: Create potential solutions to solve hardware and software problems using common troubleshooting strategies.	✓
5.NI.1: Model how information is broken down into smaller pieces, transmitted as packets (data groups) through multiple devices over networks and the Internet, and reassembled at the destination.	✓
5.DA.1: Explain how the amount of space required to store data differs based on the type of data and level of detail and that the utility of that data varies.	✓
5.DA.2: Organize and share collected data visually to highlight relationships and support a claim.	✓
5.DA.3: Prioritize, analyze and use data to communicate ideas, highlight relationships and predict outcomes.	✓
5.AP.1: Compare and refine multiple algorithms for the same task and determine which is the most appropriate.	✓
5.AP.2: Decompose problems into smaller, manageable tasks which may themselves be deconstructed and analyzed.	✓
5.AP.3: Create programs by incorporating smaller portions of existing programs, to develop something new or add more advanced features.	✓
5.AP.4: Use an iterative process to plan and develop a program by considering the perspectives and preferences of others.	✓
5.AP.5: Recognize and observe intellectual property rights and give appropriate attribution when creating, remixing, or combining programs.	✓
5.AP.6: Describe choices made during program development using code comments, presentations, and demonstrations.	✓
5.IC.1: Propose ways to improve the accessibility and usability of technology products for the diverse needs and wants of users.	✓
5.IC.2: Seek and explain the impact of diverse perspectives for the purpose of improving computational artifacts.	✓
5.CT.1: Develop algorithms in computer programs to solve problems, including unique and repeated sub-tasks within a larger program.	✓

6th Grade

Standard	Status (7/7, 100%)
6.CS.1: Utilize troubleshooting strategies to resolve hardware and software issues in a logical order.	✓
6.NI.1: Explain potential security threats and practice protective measures to reduce these threats.	✓
6.DA.1: Represent a single data set in multiple ways using words, symbols, manipulatives, charts, diagrams, and visuals.	✓
6.AP.1: Design and illustrate algorithms to efficiently solve complex problems by utilizing pseudocode and/or other descriptive methods.	✓
6.AP.2: Create naming conventions for variables that support the debugging process and incorporate these variables into a simple program.	✓
6.AP.3: Annotate programs in order to document their use and improve readability, testing, and debugging.	✓
6.IC.1: Recognize and discuss issues of bias and accessibility in existing technologies.	✓

7th Grade

Standard	Status
	(8/8, 100%)
7.CS.1: Design modifications to computing devices in order to improve the ways users interact with the devices.	✓
7.NI.1: Model the role of protocols in transmitting data across networks and the Internet.	✓
7.DA.1: Collect data using computational tools and transform the data to make it more useful.	✓
7.AP.1: Design and iteratively develop programs that combine control structures.	✓
7.AP.2: Seek and incorporate feedback from team members and users to refine a solution to a programming project that meets the user's needs.	✓
7.AP.3: Systematically test and refine programs using a range of test cases.	✓
7.AP.4: Select and assign tasks to maintain a project timeline when collaboratively developing computational artifacts.	✓
7.IC.1: Compare tradeoffs associated with computing technologies that affect people's everyday activities and career options.	✓

8th Grade

Standard	Status
	(6/6, 100%)
8.CS.1: Design a project that combines hardware and software components to collect and exchange data.	✓
8.NI.1: Explain how proper protocols transmit data across networks and the internet.	✓
8.DA.3: Test and analyze the effects of changing variables in models/simulations.	✓
8.AP.1: Develop a program with iterative protocols that combine control structures and use compound conditions.	✓
8.AP.2: Create procedures with or without parameters to organize code and make it easier to reuse.	✓
8.AP.3: Create a new program incorporating existing code, media, and libraries; and give proper attribution.	✓

Skill Struck

9th & 10th Grade

Standard	Status (20/20, 100%)
9/10.CS.1: Describe ways in which the specific implementation details of a computing system are hidden by abstractions in order to manage complexity.	✓
9/10.CS.2: Identify the different levels of abstraction in a computer system.	✓
9/10.CS.3: Develop guidelines that communicate systematic troubleshooting strategies that others can use to identify and fix errors.	✓
9/10.NI.1: Describe essential elements for connecting to a network and identify issues that impact network functionality.	✓
9/10.NI.2: Describe the design structure of the internet and identify standard protocols.	✓
9/10.DA.1: Demonstrate different representations of data (numbers, characters, and images).	✓
9/10.DA.2: Describe disadvantages or benefits associated with how data elements are organized and stored.	✓
9/10.DA.3: Create data visualizations to help others better understand real-world phenomena or factual data information.	✓
9/10.AP.1: Design algorithms to solve computational problems using a combination of original and existing algorithms.	✓
9.10.AP.2: Create more generalized computational solutions using collections of items (like an array or list) instead of separating using individual items.	✓
9/10.AP.3: Decompose problems into multiple smaller problems through systematic analysis, using constructs (such as procedures, modules, functions, methods, and/or classes).	✓
9/10.AP.4: Create computational artifacts using modular design.	✓
9/10.AP.5: Identify and collaboratively suggest changes to an application's design using feedback from a variety of users.	✓

9th & 10th Grade

Standard	Status
9/10.AP.6: Explain the limitations of licenses that restrict computational artifacts when using resources created by others.	✓
9/10.AP.7: Iteratively evaluate and refine a computational artifact to enhance its performance, reliability, usability, and accessibility.	✓
9/10.AP.8: Design and develop computational artifacts using collaborative tools.	✓
9/10.AP.9: Create documentation (pseudocode) that communicates the design of the solution to a complex problem using text, graphics, and/or demonstrations.	✓
9/10.IC.1: Evaluate how computing has impacted and/or impacts personal, ethical, social, economic, and cultural practices.	✓
9/10.IC.2: Understand that bias is always introduced when creating computational artifacts, identify ways that this unintended bias may impact equity, and then evaluate methods for alleviating that impact.	✓
9/10.IC.3: Identify solutions to problems in other content areas using established algorithms.	✓

11th & 12th Grade

Standard	Status
	(10/10, 100%)
11/12.NI.1: Identify types of security threats, and then compare and contrast measures that can be used to address, resolve, and/or prevent identified threats.	✓
11/12.NI.2: Compare and contrast cryptographic techniques to model the secure transmission of information (data).	✓
11/12.DA.1: Refine or create computational artifacts to better represent the relationships among different elements of data collected from factual sources or other processes.	✓
11/12.AP.1: Iteratively design and develop computational artifacts for practical, personal, or societal expression that implements an algorithm based on the result of an evaluation or user input.	✓
11.12.AP.2: Systematically design and create programs for broad audiences by incorporating feedback from users.	✓
11/12.AP.3: Design and develop computational artifacts working in team roles using collaborative tools.	✓
11/12.AP.4: Produce documentation to support the decisions made during the design and creation process using text, graphics, presentations, and/or demonstrations in the development of complex programs.	✓
11/12.IC.1: Evaluate and discuss the ways computing impacts personal, ethical, social, economic, and cultural practices.	✓
11/12.IC.2: Identify impacts of bias and equity deficits on design and implementation of computational artifacts, while evaluating appropriate processes for identifying issues of bias.	✓
11/12.IC.3: Demonstrate computational thinking using algorithms to problem solving across multiple disciplines.	✓