vidcode

Wyoming State CS Standards Alignment

Vidcode meets high school students where they are, encouraging them to connect computing to their daily life and the way they experience technology. The lessons are designed for a flipped classroom, where students are exposed to the material through self-paced homework tutorials. Practice and iteration takes place during class time, when learners can fully leverage the full range of instructional materials, references, peers, and teachers, as well as experiencing the authentic process of software development.

Students learn through coursework that focus on creativity and social impact. They collaborate to create digital artifacts including videos, apps, games, and interactive exhibits from day one, using programming as a tool to share their research and point of view.

The activities in each unit allow students to practice using these computer science and programming concepts in creative projects that can be as exploratory or academic as needed.

Tutorials / Code challenges
Tutorials are the main vehicle of instruction. They introduce new concepts and allow students to begin practicing them. They also review previously learned concepts and relate them to new ones.

Practices/ Code challenges + Concepts reviewed
Practices review concepts from the tutorials and allow students to spend more time implementing concepts in engaging projects. Vidcode curriculum promotes spiral design for student learning. Practices are optional if classes are pressed for time.

Quizzes
Quizzes are provided to help instructors make sure students have a good grasp on key concepts. They are available in the online Vidcode units and a summative assessment is provided for the end of the course.

Final Projects
The final projects of Creative Coding 1 have students research a range of topics related to the internet and present their findings to their classmates. Students are also expected to write two sample AP CS Principles exam questions related to their topic.

Sharing and Reflection
Each tutorial and practice includes suggestions for how students can share projects, as well as reflection questions that teachers and students can discuss as a class or develop into a writing activity.

vidcode

Vidcode's curriculum aligns with the following standards:

GRADES 3-5

Students learn the basics of JavaScript programming through the creation of memes, video projects and animations. Variables, arrays, loops, conditionals, and randomness are covered. They use what they learn to produce informative projects that communicate about the history and structure of the Internet. At this grade level, there's a focus on introductory concepts and connecting computer programming to other subject areas.

Standards	Category	Supporting Vidcode Units
5.CS.D.01 Independently, describe how internal and external parts of computing devices function to form a system	**Computing Systems** Devices	Abstraction and Global Impact: Make it Interactive
5.CS.HS.01 Model how information is translated, transmitted, and processed in order to flow through hardware and software to accomplish tasks.	**Computing Systems** Hardware & Software	Abstraction and Global Impact: Make it Interactive
5.NI.C.01 Discuss real-world cybersecurity problems and identify and implement appropriate strategies for how personal information can be protected	**Networks & The Internet** Cybersecurity	Abstraction and Global Impact: Word Wizardry Abstraction and Global Impact: Cybersecurity
5.DA.S.01 Justify the format and location for storing data based on sharing requirements and the type of information (e.g., images, videos, text).	**Data & Analysis** Storage	Programming and the Internet: Intro to Programming
5.DA.CVT.01 Organize and present collected data to highlight relationships and support a claim.	**Data & Analysis** Collection, Visualization & Transformation	Algorithms and Data: Make Your Mark
5.DA.IM.01 Use data to highlight or propose relationships, predict outcomes, or communicate an idea.	**Data & Analysis** Inference & Models	Algorithms and Data: Make Your Mark

5.AP.A.01 Using grade appropriate content and complexity, compare and refine multiple algorithms for the same task and determine which is the most appropriate.	**Algorithms & Programming** Algorithms	Abstraction and Global Impact: Algorithms and Art
5.AP.V.01 Using grade appropriate content and complexity, create programs that use variables to store and modify data.	**Algorithms & Programming** Variables	Programming and the Internet: Intro to Programming
5.AP.C.01 Using grade appropriate content and complexity, create programs that include sequences, events, loops, and conditionals, both individually and collaboratively.	**Algorithms & Programming** Control	Programming and the Internet: Intro to Programming Programming and the Internet: Loops and Animations Programming and the Internet: Conditional Logic and Special Effects
5.AP.M.01 Using grade appropriate content and complexity, decompose (break down) problems into smaller, manageable subproblems to facilitate the program development process.	**Algorithms & Programming** Modularity	Programming and the Internet: Intro to Programming
5.AP.M.02 Using grade appropriate content and complexity, modify, remix, or incorporate portions of an existing program into one's own work, to develop something new or add more advanced features.	**Algorithms & Programming** Modularity	Programming and the Internet: Conditional Logic and Special Effects
5.AP.PD.01 Use an iterative process to plan the development of a program by including others' perspectives and considering user preferences.	**Algorithms & Programming** Program Development	Abstraction and Global Impact: Interactivity Abstraction and Global Impact: Video Game Development
5.AP.PD.02 Using grade appropriate content and complexity, observe intellectual property rights and give appropriate credit when creating or remixing programs	**Algorithms & Programming** Program Development	Programming and the Internet: Intro to Programming

5.AP.PD.03 Using grade appropriate content and complexity, test and debug (i.e., identify and fix errors) a program or algorithm to ensure it runs as intended.	**Algorithms & Programming** Program Development	Programming and the Internet: Intro to Programming
5.AP.PD.04 Using grade appropriate content and complexity, describe choices made during program development using code comments, presentations, and demonstrations.	**Algorithms & Programming** Program Development	Programming and the Internet: Intro to Programming Programming and the Internet: Arrays Programming and the Internet: Loops and Animations Programming and the Internet: Conditional Logic and Special Effects
5.AP.PD.05 Using grade appropriate content and complexity, with teacher guidance, perform varying roles when collaborating with peers during the design, implementation, and review stages of program development.	**Algorithms & Programming** Program Development	Programming and the Internet: Intro to Programming
5.IC.C.01 Give examples and explain how computing technologies have changed the world and express how those technologies influence and are influenced by cultural practices	**Impacts of Computing** Culture	Programming and the Internet: Intro to Programming
5.IC.SI.01 Seek diverse perspectives for the purpose of improving computational artifacts	**Impacts of Computing** Social Interactions	Algorithms and Data: Make Your Mark Algorithms and Data: Make Up Your Mind
5.IC.SLE.01 Recognize and appropriately use public domain and creative commons media and discuss the social impact of violating intellectual property rights.	**Impacts of Computing** Safety, Law, & Ethics	Programming and the Internet: Intro to Programming

GRADES 6-8

Students build on the concepts covered in 3-5 with events, for and while loops, nested control structures, string methods, and object constructors. They learn about complex applications of JavaScript programming including interactivity, algorithms, and procedural art. Special focus is also given to the importance of collaboration and building a supportive programmer community, and the final projects reflect on the impact of computing on others.

Standards	Category	Supporting Vidcode Units
8.CS.T.01 Systematically identify, resolve, and document increasingly complex software and hardware problems with computing devices and their components	**Computing Systems** Troubleshooting	Software Development and Cybersecurity
8.NI.NCO.01 Model the role of protocols in transmitting data across networks and the internet (e.g., explain protocols and their importance to data transmission; model how packets are broken down into smaller pieces and how they are delivered).	**Networks & The Internet** Network Communication & Organization	Programming and the Internet
8.NI.C.01 Critique physical and digital procedures that could be implemented to protect electronic data/information.	**Networks & The Internet** Cybersecurity	Software Development and Cybersecurity
8.NI.C.02 Apply multiple methods of encryption to model the secure transmission of data.	**Networks & The Internet** Cybersecurity	Software Development and Cybersecurity
8.DA.S.01 Represent data using multiple encoding schemes (e.g., ASCII, binary).	**Data & Analysis** Storage	Data Visualization Software Development and Cybersecurity
8.DA.CVT.01 Using computational tools, transform collected data to make it more useful and reliable.	**Data & Analysis** Collection, Visualization, & Transformation	Data Visualization

8.AP.A.01 Create flowcharts and pseudocode to design algorithms to solve complex problems.	**Algorithms & Programming** Algorithms	Algorithms and Data: Make it Click! Algorithms and Data: Make Your Mark
8.AP.V.01 Using grade appropriate content and complexity, create clearly named variables that represent different data types and perform operations on their values	**Algorithms & Programming** Variables	Programming Logic and Global Connection
8.AP.C.01 Using grade appropriate content and complexity, design and iteratively develop programs that combine control structures, including nested loops and compound conditionals.	**Algorithms & Programming** Control	Programming and the Internet: Conditional Logic and Special Effects Abstraction and Global Impact: Algorithms and Art
8.AP.M.01 Using grade appropriate content and complexity, decompose problems and subproblems into parts to facilitate the design, implementation, and review of programs.	**Algorithms & Programming** Modularity	Programming Logic and Global Connection
8.AP.M.02 Using grade appropriate content and complexity, create procedures with parameters to organize code and make it easier to reuse.	**Algorithms & Programming** Modularity	Programming Logic and Global Connection
8.AP.PD.01 Using grade appropriate content and complexity, seek and incorporate feedback from team members and users to refine a solution to a problem.	**Algorithms & Programming** Program Development	Programming Logic and Global Connection
8.AP.PD.02 Incorporate existing code, media, and libraries into original programs of increasing complexity and give attribution.	**Algorithms & Programming** Program Development	Programming Logic and Global Connection

8.AP.PD.03 Systematically test and refine programs using a range of test cases.	**Algorithms & Programming** Program Development	Programming Logic and Global Connection Software Development
8.AP.PD.04 Using grade appropriate content and complexity, document programs in order to make them easier to follow, test, and debug.	**Algorithms & Programming** Program Development	Programming Logic and Global Connection Software Development
8.AP.PD.05 Distribute tasks and maintain a project timeline when collaboratively developing computational artifacts.	**Algorithms & Programming** Program Development	Global Impacts of Computer Science Software Development
8.IC.C.02 Describe issues of bias and accessibility in the design of technologies.	**Impacts of Computing** Culture	Global Impacts of Computer Science
8.IC.SI.01 Using grade appropriate content and complexity, collaborate using tools to connect with peers when creating a computational artifact.	**Impacts of Computing** Social Interactions	Software Development
8.IC.SLE.02 Using grade level appropriate content and complexity, discuss the legal, social, and ethical impacts associated with software development and use, including both positive and malicious intent.	**Impacts of Computing** Safety, Law, & Ethics	Programming and the Internet Global Impacts of Computer Science

HIGH SCHOOL LEVEL 1

At the high school level, classes are transformed into a software development team. The projects cover a wide range of visual, interactive and algorithmic elements that students can recombine into useful apps, including custom buttons and sliders, spawning multiple copies of objects, managing large numbers of variables in data structures, and writing readable and reusable code. Students work together to create meaningful computational artifacts that communicate information and solve problems.

Standards	Category	Supporting Vidcode Units
L1.CS.D.01 Explain how abstractions hide the underlying implementation details of computing systems embedded in everyday objects.	**Computing Systems** Devices	Algorithms and Data
L1.NI.C.01 Give examples to illustrate how sensitive data can be affected by malware and other attacks.	**Networks & The Internet** Cybersecurity	Software Development and Cybersecurity
L1.NI.C.02 Recommend cybersecurity measures to address various scenarios based on factors such as efficiency, feasibility, and ethical impacts.	**Networks & The Internet** Cybersecurity	Software Development and Cybersecurity
L1.NI.C.03 Compare various security measures, considering trade-offs between the usability and security of a computing system.	**Networks & The Internet** Cybersecurity	Software Development and Cybersecurity
L1.NI.C.04 Explain trade-offs when selecting and implementing cybersecurity recommendations.	**Networks & The Internet** Cybersecurity	Software Development and Cybersecurity
L1.DA.S.01 Translate between different bit representations of real-world phenomena, such as characters, numbers, and images.	**Data & Analysis** Storage	Data Visualization Developing Applications for Everyone

L1.DA.CVT.01 Create interactive data representations using software tools to help others better understand real-world phenomena (e.g., paper surveys and online data sets).	**Data & Analysis** Collection, Visualization, & Transformation	Data Visualization
L1.DA.IM.01 Create computational models that represent the relationships among different elements of data collected from a phenomenon or process.	**Data & Analysis** Inference & Models	Data Visualization
L1.AP.A.01 Create a prototype that uses algorithms (e.g., searching, sorting, finding shortest distance) to provide a possible solution for a real-world problem relevant to the student.	**Algorithms & Programming** Algorithms	Programming Fundamentals with JavaScript
L1.AP.A.02 Describe how artificial intelligence algorithms drive many software and physical systems.	**Algorithms & Programming** Algorithms	Programming Fundamentals with JavaScript Software Development and Cybersecurity
L1.AP.V.01 Use lists to simplify solutions, generalizing computational problems instead of repeatedly using simple variables.	**Algorithms & Programming** Variables	Programming Fundamentals with JavaScript
L1.AP.C.01 Justify the selection of specific control structures when tradeoffs involve implementation, readability, and program performance, and explain the benefits and drawbacks of choices made.	**Algorithms & Programming** Control	Programming Fundamentals with JavaScript Developing Applications for Everyone
L1.AP.C.02 Trace the execution of loops and conditional statements, illustrating output and changes in values of named variables.	**Algorithms & Programming** Control	Programming Fundamentals with JavaScript

L1.AP.C.03 Design and iteratively develop computational artifacts for practical intent, personal expression, or to address a societal issue by using events to initiate instructions.	**Algorithms & Programming** Control	Programming Fundamentals with JavaScript Software Development and Cybersecurity
L1.AP.M.01 Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects.	**Algorithms & Programming** Modularity	Programming Fundamentals with JavaScript Developing Applications for Everyone
L1.AP.M.02 Create artifacts by using procedures within a program, combinations of data and procedures, or independent but interrelated programs.	**Algorithms & Programming** Modularity	Software Development and Cybersecurity
L1.AP.PD.01 Plan and develop programs by analyzing a problem and/or process, developing and documenting a solution, testing outcomes, and adapting the program for a variety of users.	**Algorithms & Programming** Program Development	Software Development and Cybersecurity
L1.AP.PD.03 Use debugging tools to identify and fix errors in a program.	**Algorithms & Programming** Program Development	Programming Fundamentals with JavaScript
L1.AP.PD.04 Design and develop computational artifacts, working in team roles, using collaborative tools.	**Algorithms & Programming** Program Development	Developing Applications for Everyone Software Development and Cybersecurity
L1.AP.PD.06 Evaluate and refine computational artifacts to make them more usable and accessible.	**Algorithms & Programming** Program Development	Developing Applications for Everyone
L1.IC.C.02 Test and refine computational artifacts to reduce bias and equity deficits.	**Impacts of Computing** Culture	Developing Applications for Everyone

L1.IC.C.03 Demonstrate how a given algorithm applies to problems across disciplines.	**Impacts of Computing** Culture	Cross-curricular Programming
L1.IC.SI.01 Use tools and methods for collaboration.	**Impacts of Computing** Social Interactions	Software Development and Cybersecurity
L1.IC.SLE.03 Evaluate the social and economic implications of privacy in the context of safety, law, or ethics.	**Impacts of Computing** Safety, Law, & Ethics	Software Development and Cybersecurity
L1.IC.SLE.04 Using grade level appropriate content and complexity, discuss the legal, social, and ethical impacts associated with software development and use, including both positive and malicious intent.	**Impacts of Computing** Safety, Law, & Ethics	Software Development and Cybersecurity

HIGH SCHOOL LEVEL 2

Standards	Category	Supporting Vidcode Units
L2.NI.C.01 Compare ways software developers protect devices and information from unauthorized access.	**Networks & The Internet** Cybersecurity	Software Development and Cybersecurity
L2.DA.CVT.01 Use data analysis tools and techniques to identify patterns in data representing complex systems.	**Data & Analysis** Collection, Visualization, & Transformation	Data Visualization Developing Applications for Everyone
L2.DA.CVT.02 Select data collection tools and techniques, and use them to generate data sets that support a claim or communicate information.	**Data & Analysis** Collection, Visualization, & Transformation	Data Visualization Developing Applications for Everyone
L2.DA.IM.01 Formulate, refine, and test scientific hypotheses using models and simulations.	**Data & Analysis** Inference & Models	Data Visualization
L2.AP.A.01 Critically examine and trace classic algorithms. Use and adapt classic algorithms to solve computational problems (e.g., selection sort, insertion sort, binary search, linear search).	**Algorithms & Programming** Algorithms	Programming Fundamentals with JavaScript Data Visualization
L2.AP.M.01 Construct solutions to problems using student-created components, such as procedures, modules, and/or objects.	**Algorithms & Programming** Modularity	Software Development and Cybersecurity
L2.AP.PD.01 Plan and develop programs that will provide solutions to a variety of users using a software life cycle process.	**Algorithms & Programming** Program Development	Software Development and Cybersecurity

L2.AP.PD.04 Evaluate key qualities of a program through a process such as a code review (e.g., qualities could include correctness, usability, readability, efficiency, portability, and scalability).	**Algorithms & Programming** Program Development	Software Development and Cybersecurity
L2.AP.PD.06 Explain security issues that might lead to compromised computer programs.	**Algorithms & Programming** Program Development	Software Development and Cybersecurity
L2.AP.PD.07 Modify an existing program to add additional functionality and discuss intended and unintended implications (e.g., breaking other functionality).	**Algorithms & Programming** Program Development	Programming Fundamentals with JavaScript Developing Applications for Everyone
L2.IC.C.02 Evaluate the impact of equity, access, and influence on the distribution of computing resources in a global society.	**Impacts of Computing** Culture	Developing Applications for Everyone
L2.IC.SLE.02 Using grade level appropriate content and complexity, discuss the legal, social, and ethical impacts associated with software development and use, including both positive and malicious intent.	**Impacts of Computing** Safety, Law, & Ethics	Software Development and Cybersecurity