Interdisciplinary Computer Science
Fifth Grade Course Syllabus
One Year for Elementary School

Course Overview and Goals

The Interdisciplinary Computer Science Fifth Grade Course introduces students to foundational programming and computational thinking through a block-based programming environment. Students apply coding concepts across mathematics, science, English language arts (ELA), and social studies while developing problem-solving and critical thinking skills.

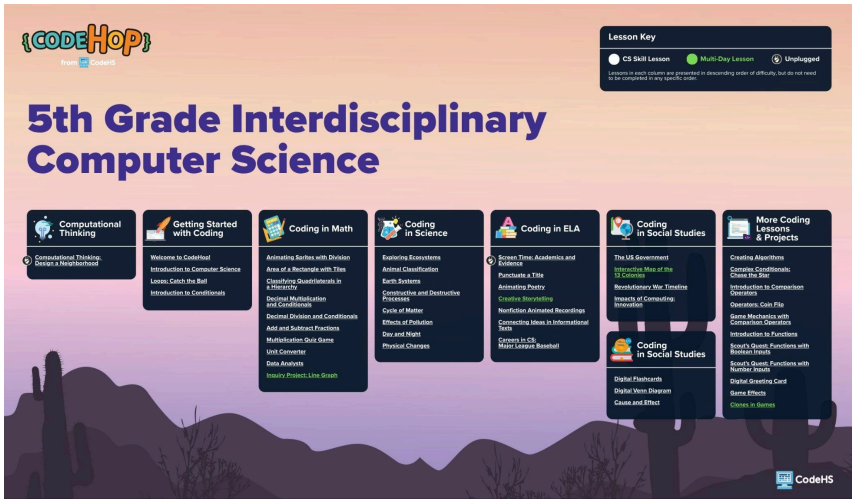
Learning Environment: This teacher-led course includes ready-to-use lesson plans that follow a consistent instructional format: Review, Guided Practice, Independent Practice, Extension, and Reflection. The course begins with the Computational Thinking module, followed by the Getting Started with Coding unit. After completing these introductory units, teachers may teach lessons from the mathematics, science, English language arts (ELA), social studies, Cross-Curricular, and More Coding sections in the order that best supports their instructional goals. The Cross-Curricular section includes coding lessons that can be integrated into any subject area. Lessons within each section progress in approximate order of complexity, and spiral review opportunities help reinforce previously learned concepts.

Programming Environment: Students write and run programs within the CodeHop platform using a block-based programming environment.

Assessments: Some units conclude with a brief five-question quiz designed to assess student understanding of the concepts and skills introduced in the unit. Quizzes can be assigned digitally, completed as a QuizBuzz activity, or printed for paper-based completion.

Prerequisites: There are no prerequisites for this course. It is designed to support all learners, regardless of prior computer science experience.

More Information: Browse the content of this course at https://codehs.com/course/20694/overview



A clickable PDF can be found at https://codehs.com/InterCSRoadmaps

Course Breakdown

Computational Thinking

Students start the course by applying computational thinking skills to a community design challenge. They will identify patterns, break down big ideas into smaller steps, sequence actions, and create a plan to bring their neighborhood designs to life.

Objectives / Topics Covered	● Applying computational thinking to solve a design challenge ● Sequencing and organizing steps to complete a project ● Recognizing patterns and simplifying processes
Lessons	**Computational Thinking: Design a Neighborhood** ● Use computational thinking to design a neighborhood by identifying patterns, sequencing instructions, and simplifying processes. **Computational Thinking: Design a Neighborhood Quiz (5 questions)**

Getting Started with Coding

Students learn how to access CodeHop and start programming. They will be introduced to loops, events, and conditionals while creating simple interactive programs.

Objectives / Topics Covered	● Navigating the CodeHop Playground ● Using loops and conditionals ● Understanding and applying basic programming concepts
Lessons	**Welcome to CodeHop! (15 minute lesson)** ● Learn how to log in and use the CodeHop Playground. **Introduction to Computer Science** ● Define key CS terms and build a simple program. **Loops: Catch the Ball** ● Use two types of loops to create a simple game. **Introduction to Conditionals** ● Understand what conditionals are and create a program using if/then blocks. **Getting Started with Coding Quiz (5 questions)**

Coding in Math

Students connect math skills to coding by creating programs that work with division, fractions, decimals, measurement conversions, geometry, and data visualization.

Objectives / Topics Covered	● Modeling math operations and problem-solving with code ● Using variables, operators, loops, and conditionals in math programs ● Creating interactive calculators and quizzes ● Visualizing math concepts with animations and graphs
Lessons	**Animating Sprites with Division** ● Animate sprites with loops and wait blocks to show division. **Area of a Rectangle with Tiles** ● Model fractional area using loops and variables. **Classifying Quadrilaterals in a Hierarchy** ● Categorize quadrilaterals and document with comments. **Decimal Multiplication and Conditionals** ● Use conditionals to review decimal multiplication.

	Decimal Division and Conditionals ● Create an interactive decimal division game with levels. **Add and Subtract Fractions** ● Create a fractions quiz using broadcast messages and patterns. **Unit Converter** ● Program a measurement converter with variables and conditionals. **Multiplication Quiz Game** ● Create a multiplication quiz game using variables, operators, conditionals, and loops. **Data Analysts** ● Analyze data to draw conclusions and make predictions. **Inquiry Project: Line Graph** ● Display data investigation results in a line graph.

Coding in Science

Students explore ecosystems, classification, Earth processes, cycles, and environmental impacts by creating animated models and simulations in CodeHop.

Objectives / Topics Covered	● Modeling science processes and systems with code ● Using events, loops, and conditionals to simulate real-world concepts ● Creating interactive visual explanations of science topics
Lessons	**Exploring Ecosystems** ● Model the flow of energy in an ecosystem. **Animal Classification** ● Create a sorting program for animal groups. **Earth Systems** ● Show wind and rain interactions with the geosphere. **Constructive and Destructive Processes** ● Animate volcanic changes to Earth's surface. **Cycle of Matter** ● Animate matter cycles using events and messages. **Effects of Pollution** ● Use conditionals to demonstrate how actions affect water pollution. **Day and Night** ● Show Earth's rotation causing day/night using comparison operators. **Physical Changes** ● Use variables and conditionals to simulate phase changes of matter.

Coding in ELA

Students apply coding skills to literacy concepts, creating interactive activities, animated readings, and storytelling projects.

Objectives / Topics Covered	● Using conditionals and operators in language-based games ● Animating poetry, nonfiction, and creative writing ● Connecting reading comprehension with programming skills
Lessons	**Screen Time: Academics and Evidence** ● Create a healthy screen-time plan and write an opinion about screen-time rules supported with reasons. **Punctuate a Title** ● Make a game that teaches title punctuation using conditionals. **Animating Poetry** ● Animate and narrate poetry for meaning and tone.

	Creative Storytelling ● Plan and animate an original story with events and sequences. **Nonfiction Animated Recordings** ● Build an animated reading of nonfiction text. **Connecting Ideas in Informational Texts** ● Create a program that demonstrates the relationships between key ideas or events from an informational text. **Careers in CS: Major League Baseball** ● Show how coding is used in sports through an animated timeline.

Coding in Social Studies

Students use CodeHop to model civics, historical timelines, vocabulary review, and map-based interactivity.

Objectives / Topics Covered	● Using coding to present social studies content interactively ● Applying conditionals, variables, and events to history and geography projects
Lessons	**The US Government** ● Model how a bill becomes a law with if/else conditionals. **Interactive Map of the 13 Colonies** ● Create a map-based program with events and variables. **Revolutionary War Timeline** ● Build an interactive historical timeline. **Impacts of Computing: Innovation** ● Compare past and present versions of technology and predict impacts on different communities.

Cross-Curricular

Apply coding skills across multiple subjects with these versatile project templates.

Objectives / Topics Covered	● Using broadcasts and messages for interactivity ● Implementing logic to show relationships and comparisons
Lessons	**Digital Flashcards** ● Use broadcast messages to build a set of digital flashcards for any subject. **Digital Venn Diagram** ● Program an interactive Venn diagram to compare two things. **Cause and Effect** ● Create a program that demonstrates a cause-and-effect relationship using messages and conditionals.

More Coding Lessons & Projects

Students expand their programming skills with advanced logic, comparison operators, functions, and game design.

Objectives / Topics Covered	● Applying complex logic and operators in programs ● Creating reusable code with functions ● Enhancing games and projects with effects, clones, and custom features
Lessons	**Creating Algorithms** ● Compare multiple algorithms to find the best solution.

	Complex Conditionals: Chase the Star ● Use if/then/else logic in a game. **Introduction to Comparison Operators** ● Use operators in if/else statements. **Operators: Coin Flip** ● Simulate a coin flip with variables and operators. **Game Mechanics with Comparison Operators** ● Create game-ending mechanics. **Introduction to Functions** ● Build and use functions. **Scout's Quest: Functions with Boolean Inputs** ● Use boolean inputs in functions for different outcomes. **Scout's Quest: Functions with Number Inputs** ● Draw using functions with numeric inputs. **Digital Greeting Card** ● Create an animated greeting card with loops and events. **Game Effects** ● Add visual effects and polish to a game. **Clones in Games** ● Use clones to create an endless runner game.

Interdisciplinary Computer Science Fifth Grade Course Supplemental Materials

Resources	Description
Parent Welcome Letter (Spanish)	Send this letter home to introduce families to their new computer science curriculum.
Warm-Up Activities	This warm-up activity slide deck provides 5-10 minute problems aligned with computer science skills to engage students at the start of class, allowing teachers to preview or review concepts with answer keys and discussion tips included in the Speaker Notes.
Program Self-Assessment (Spanish)	This is a student self-assessment tool designed to help K-6 learners reflect on their programming projects, evaluate their skills in algorithms, debugging, collaboration, and reflection, and set goals for improvement.
Peer Review Resources (Spanish)	This provides structured worksheets to facilitate student feedback during collaborative coding projects. It encourages reflection by guiding students to highlight successes, ask questions, and offer constructive feedback on their partner's work.
Lesson Reflection & Computational Thinking (Spanish)	This guides students in engaging with computational thinking concepts, preparing for discussions, reflecting on lessons, and applying their learning to real-world problem-solving.
Design-Your-Own-Lesson Templates	Empower your students to explore and express their knowledge creatively with our versatile graphic organizer templates. Designed with adaptability and ease of use in mind, these interactive tools transform any subject into an engaging, hands-on learning experience.

These resources and more are found on the **CodeHop Resources Page**.