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

Course Overview and Goals

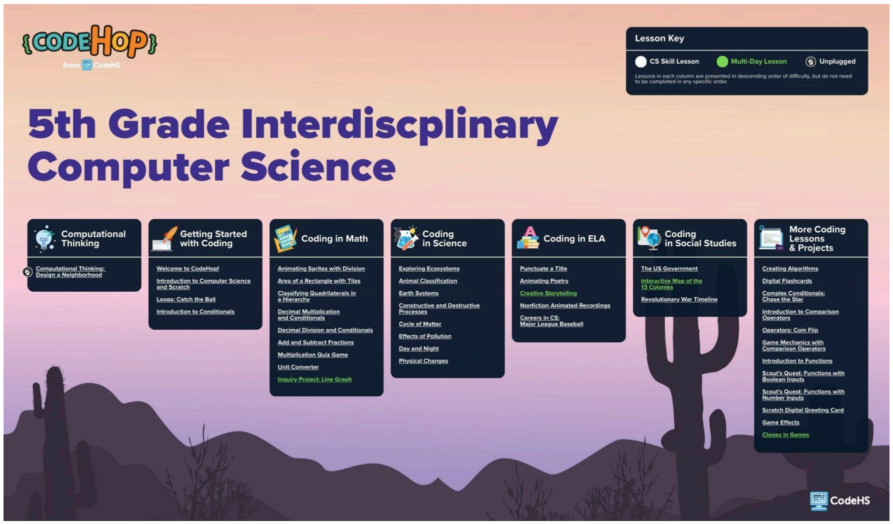
The Interdisciplinary Computer Science Fifth Grade Course introduces students to foundational programming and computational thinking concepts through **Scratch**, a block-based programming language designed for early learners. Students will develop coding skills while applying them across math, science, English language arts (ELA), and social studies. A dedicated **Computational Thinking** module reinforces practices such as decomposition, pattern recognition, abstraction, and algorithm design, helping students strengthen their problem-solving and critical thinking abilities in every subject area.

Learning Environment: This course is designed to be teacher-led, with ready-to-use lesson plans that follow a structured format: Review, Guided Practice, Independent Practice, Extension, and Reflection. The course begins with a Computational Thinking module, which can be taught on its own or integrated into other subjects, followed by the Getting Started with Coding unit to establish basic programming skills. Teachers can then choose from subject-area coding lessons in any order, with topics arranged in approximate order of complexity to support flexible pacing. With **more than 36 lessons**, teachers can select activities that align with their instructional goals, while spiral review helps reinforce learning. The **"I do, we do, you do"** model supports gradual release of responsibility, building student independence and confidence.

Programming Environment: Students will write and run programs in Scratch within the CodeHS platform. This environment supports hands-on programming, enabling students to design, test, and debug interactive projects in a user-friendly space.

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



Lesson Key

○ CS Skill Lesson ● Multi-Day Lesson Unplugged

Lessons in each column are presented in descending order of difficulty, but do not need to be completed in any specific order.

5th Grade Interdiscplinary Computer Science

Computational Thinking

Computational Thinking: Design a Neighborhood

Getting Started with Coding

Welcome to CodeHop!

Introduction to Computer Science and Scratch

Loops: Catch the Ball

Introduction to Conditionals

Coding in Math

Animating Sprites with Division

Area of a Rectangle with Tiles

Classifying Quadrilaterals in a Hierarchy

Decimal Multiplication and Conditionals

Decimal Division and Conditionals

Add and Subtract Fractions

Multiplication Quiz Game

Unit Converter

Inquiry Project: Line Graph

Coding in Science

Exploring Ecosystems

Animal Classification

Earth Systems

Constructive and Destructive Processes

Cycle of Matter

Effects of Pollution

Day and Night

Physical Changes

Coding in ELA

Punctuate a Title

Animating Poetry

Creative Storytelling

Nonfiction Animated Recordings

Careers in CS: Major League Baseball

Coding in Social Studies

The US Government

Interactive Map of the 13 Colonies

Revolutionary War Timeline

More Coding Lessons & Projects

Creating Algorithms

Digital Flashcards

Complex Conditionals: Chase the Star

Introduction to Comparison Operators

Operators: Coin Flip

Game Mechanics with Comparison Operators

Introduction to Functions

Scout's Quest: Functions with Boolean Inputs

Scout's Quest: Functions with Number Inputs

Scratch Digital Greeting Card

Game Effects

Clones in Games

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.

Getting Started with Coding

Students learn how to access CodeHop and start programming in Scratch. 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 in Scratch ● 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 and Scratch** ● Define key CS terms and build a simple Scratch 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.

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. **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 Scratch.

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	**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. **Careers in CS: Major League Baseball** - Show how coding is used in sports through an animated timeline.

Coding in Social Studies

Students use Scratch 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.

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. **Digital Flashcards** ● Use broadcasts to make digital flashcards. **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 in Scratch. **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. **Scratch 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 Scratch Templates	Empower your students to explore and express their knowledge creatively with our versatile Scratch 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**.	