Interdisciplinary Computer Science Kindergarten Course Syllabus

One Year for Elementary School

Course Overview and Goals

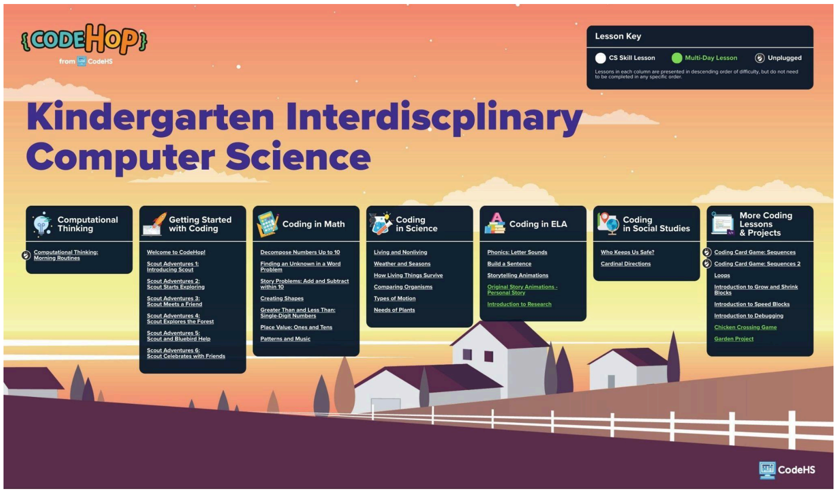
The Interdisciplinary Computer Science Kindergarten Course introduces students to foundational programming and computational thinking concepts through **ScratchJr**, a block-based programming language designed for early learners. Students will develop coding skills while applying them across math, science, 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 ScratchJr 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/20688/overview



Kindergarten Interdiscplinary Computer Science

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.

Computational Thinking
- Computational Thinking: Morning Routines

Getting Started with Coding
- Welcome to CodeHop!
- Scout Adventures 1: Introducing Scout
- Scout Adventures 2: Scout Starts Exploring
- Scout Adventures 3: Scout Meets a Friend
- Scout Adventures 4: Scout Explores the Forest
- Scout Adventures 5: Scout and Bluebird Help
- Scout Adventures 6: Scout Celebrates with Friends

Coding in Math
- Decompose Numbers Up to 10
- Finding an Unknown in a Word Problem
- Story Problems: Add and Subtract within 10
- Creating Shapes
- Greater Than and Less Than: Single-Digit Numbers
- Place Value: Ones and Tens
- Patterns and Music

Coding in Science
- Living and Nonliving
- Weather and Seasons
- How Living Things Survive
- Comparing Organisms
- Types of Motion
- Needs of Plants

Coding in ELA
- Phonics: Letter Sounds
- Build a Sentence
- Storytelling Animations
- Original Story Animations - Personal Story
- Introduction to Research

Coding in Social Studies
- Who Keeps Us Safe?
- Cardinal Directions

More Coding Lessons & Projects
- Coding Card Game: Sequences
- Coding Card Game: Sequences 2
- Loops
- Introduction to Grow and Shrink Blocks
- Introduction to Speed Blocks
- Introduction to Debugging
- Chicken Crossing Game
- Garden Project

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

Course Breakdown

Computational Thinking

Students are introduced to the core practices of computational thinking. They learn how to break down tasks, recognize patterns, put steps in order, and simplify processes. These skills will serve as a foundation for problem-solving in coding and in real-world contexts.

Objectives / Topics Covered	● Identifying and recognizing patterns ● Breaking complex tasks into smaller steps ● Sequencing actions in logical order ● Simplifying processes for efficiency
Lessons	**Computational Thinking: Morning Routines** ● Identify patterns, break down tasks, and create a step-by-step sequence for daily routines.

Getting Started with Coding

Students become familiar with the CodeHop Playground and the ScratchJr environment through a story-driven series called Scout Adventures. They learn to add characters, use motion blocks, and create interactive sequences.

Objectives / Topics Covered	● Navigating the programming environment ● Adding, modifying, and deleting characters ● Using motion blocks to move sprites ● Creating sequences of commands ● Adding backgrounds, pages, and events
Lessons	**Welcome to CodeHop! (15 minute lesson)** ● Learn how to log in and use the CodeHop Playground. **Scout Adventures 1: Introducing Scout** ● Explore the interface and add characters. **Scout Adventures 2: Scout Starts Exploring** ● Add backgrounds and a new page. **Scout Adventures 3: Scout Meets a Friend** ● Delete and modify characters. **Scout Adventures 4: Scout Explores the Forest** ● Use motion blocks to move characters. **Scout Adventures 5: Scout and Bluebird Help** ● Build sequences to collect objects. **Scout Adventures 6: Scout Celebrates with Friends** ● Create a celebration scene using multiple pages, characters, and events.

Coding in Math

Students apply coding skills to math concepts such as number decomposition, place value, patterns, and shape creation. They use events, loops, and sequences to create interactive math activities.

Objectives / Topics Covered	● Visualizing addition, subtraction, and place value ● Programming interactive math story problems ● Creating and combining shapes ● Comparing numbers and building patterns with music

Lessons	**Decompose Numbers Up to 10** ● Use events to show number decomposition. **Finding an Unknown in a Word Problem** ● Program number stories with missing values. **Story Problems: Add and Subtract within 10** ● Animate addition or subtraction problems. **Creating Shapes** ● Program shapes to respond to user interaction and combine into complex shapes. **Greater Than and Less Than: Single-Digit Numbers** ● Create a comparison program. **Place Value: Ones and Tens** ● Animate counting by ones, fives, and tens. **Patterns and Music** ● Use sequences and loops to create musical patterns.

Coding in Science

Students explore science topics through coding, modeling real-world phenomena such as weather, motion, and the needs of living things.

Objectives / Topics Covered	● Classifying living and nonliving things ● Modeling weather, seasons, and survival needs ● Comparing organisms and types of motion ● Showing plant needs using events and messages
Lessons	**Living and Nonliving** ● Classify and animate living vs. nonliving. **Weather and Seasons** ● Animate seasonal weather changes. **How Living Things Survive** ● Illustrate survival needs. **Comparing Organisms** ● Group animals by characteristics. **Types of Motion** ● Program animations for different motion types. **Needs of Plants** ● Model plant needs using events and messages.

Coding in ELA

Students strengthen literacy skills by coding interactive phonics activities, building sentences, retelling stories, and conducting research.

Objectives / Topics Covered	● Creating phonics and reading comprehension games ● Building and reading sentences aloud ● Animating story retellings and original stories ● Using research to create visual programs
Lessons	**Phonics: Letter Sounds** ● Create an interactive phonics game. **Build a Sentence** ● Program sentences that can be read aloud. **Storytelling Animations** ● Retell a story in sequence.

	Original Story Animations – Personal Story ● Animate a personal story with narration. **Introduction to Research** ● Find information and present it visually.

Coding in Social Studies

Students use coding to represent social studies concepts such as roles in the community and spatial awareness.

Objectives / Topics Covered	● Identifying authority figures and their roles ● Using cardinal directions and maps
Lessons	**Who Keeps Us Safe?** ● Animate an authority figure’s role. **Cardinal Directions** ● Create an interactive compass rose.

More Coding Lessons & Projects

Students extend their coding skills with games, special effects, and debugging activities. They explore loops, size changes, speed control, and more.

Objectives / Topics Covered	● Using loops, grow/shrink, and speed blocks ● Debugging code and adjusting game difficulty ● Creating animations with message events
Lessons	**Coding Card Game: Sequences** ● Work together to move Scout through a maze. **Coding Card Game: Sequences 2** ● Continue building sequences in a maze challenge. **Loops** ● Create programs that repeat code using loops. **Introduction to Grow and Shrink Blocks** ● Change character size dynamically. **Introduction to Speed Blocks** ● Adjust character animation speed. **Introduction to Debugging** ● Find and fix sequence errors. **Chicken Crossing Game** ● Program a player character and adjust difficulty. **Garden Project** ● Animate seed growth with message events and size changes.

Interdisciplinary Computer Science Kindergarten Course
Supplemental Materials

Resources	Description
Parent Welcome Letter (Spanish)	Send this letter home to introduce families to computer science.
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.
All of these resources and more are found on the **CodeHop Resources Page**.	