Interdisciplinary Computer Science Third Grade Course Syllabus

One Year for Elementary School

Course Overview and Goals

The Interdisciplinary Computer Science Third 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.

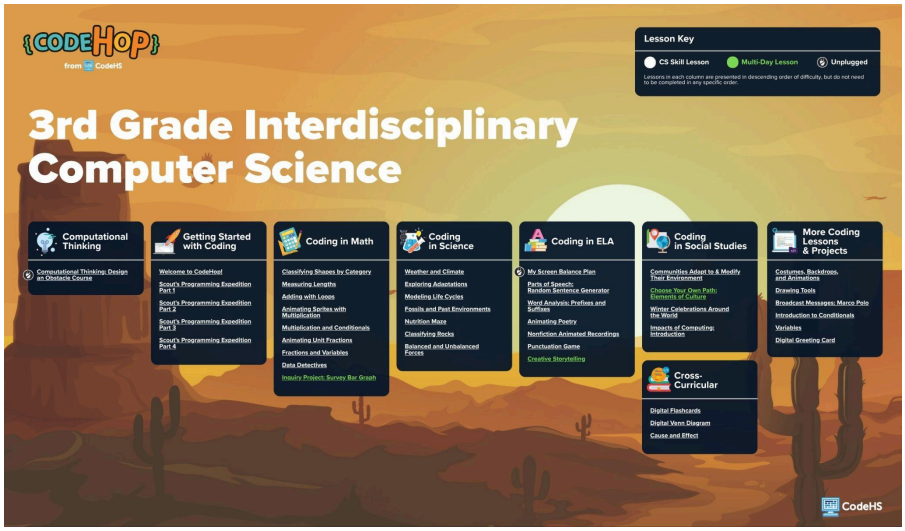
Learning Environment: This teacher-led course includes ready-to-use lesson plans that follow a consistent instructional format: Review, Guided Practice, Independent Practice, Extension, and Reflection. The course begins with the Computational Thinking module, followed by the Getting Started with Coding 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/20691/overview



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

Course Breakdown

Computational Thinking

Students are introduced to computational thinking concepts through an engaging design challenge. They will identify patterns, break down complex ideas, and sequence steps to plan and create a functional obstacle course.

Objectives / Topics Covered	● Understanding and applying computational thinking concepts ● Designing a process from idea to execution ● Sequencing steps and simplifying complex tasks
Lessons	**Computational Thinking: Design an Obstacle Course** ● Use computational thinking skills to design an obstacle course by identifying patterns, breaking down steps, and creating a step-by-step plan. **Computational Thinking: Design an Obstacle Course Quiz (5 questions)**

Getting Started with Coding

Students explore the CodeHop Playground and begin the sequential Scout's Programming Expedition series. They will learn the basics of the interface, including sprites, sequences, events, looks, and motion.

Objectives / Topics Covered	● Navigating CodeHop ● Creating and animating sprites ● Using sequences, loops, and events to tell a story
Lessons	**Welcome to CodeHop!** ● Learn how to log in and use the CodeHop Playground. **Scout's Programming Expedition Part 1** ● Use basic programming commands to program a sprite to move and talk. **Scout's Programming Expedition Part 2** ● Add sprites and create a sequence to animate a story. **Scout's Programming Expedition Part 3** ● Create an animated Scout story using loops, events, looks, and motion blocks. **Scout's Programming Expedition Part 4** ● Create an animated Scout story using events, looks, and motion blocks. **Scout's Programming Expedition Quiz (5 questions)**

Coding in Math

Students integrate math concepts with programming to create interactive projects that explore classification, measurement, operations, fractions, and variables.

Objectives / Topics Covered	● Classifying and measuring shapes ● Modeling operations with whole numbers and fractions ● Using loops, conditionals, and variables in math projects ● Creating and interpreting graphs
Lessons	**Classifying Shapes by Category** ● Create a program using events to classify quadrilaterals. **Measuring Lengths** ● Use events to create an interactive measurement game. **Adding with Loops** ● Use loops to repeat commands and add multi-digit numbers. **Animating Sprites with Multiplication**

	● Use multiplication to animate sprites with loops and wait blocks. **Multiplication and Conditionals** ● Create a program using if/then blocks to review multiplication. **Animating Unit Fractions** ● Use loops to animate repeated addition of unit fractions. **Fractions and Variables** ● Represent fractions on a number line using variables and conditionals. **Data Detectives** ● Evaluate data for reliability to draw conclusions and make predictions. **Inquiry Project: Survey Bar Graph** ● Modify a program to display the results of an investigation.

Coding in Science

Students build interactive science models to explore weather, adaptations, life cycles, nutrition, geology, and forces.

Objectives / Topics Covered	● Using events to model scientific processes ● Applying conditionals to demonstrate cause and effect ● Creating interactive representations of life, earth, and physical sciences
Lessons	**Weather and Climate** ● Use climate data and event blocks to predict weather. **Exploring Adaptations** ● Create an interactive program to show how animal adaptations aid survival. **Modeling Life Cycles** ● Program a model of a frog's life cycle using broadcast messages. **Fossils and Past Environments** ● Model fossil evidence using conditionals and loops. **Nutrition Maze** ● Create an interactive nutrition maze using conditionals. **Classifying Rocks** ● Use if/then statements to classify rocks by hardness. **Balanced and Unbalanced Forces** ● Model force impacts on speed using conditionals and variables.

Coding in ELA

Students use the CodeHop interface to enhance literacy skills by creating interactive stories, poetry animations, and grammar games.

Objectives / Topics Covered	● Building random sentence generators ● Animating poetry and nonfiction ● Creating games to practice punctuation ● Developing storytelling skills
Lessons	**My Screen Balance Plan** ● Define healthy screen time and write an opinion piece on safety. **Parts of Speech: Random Sentence Generator** ● Create simple sentences with nouns, adjectives, verbs, and adverbs. **Word Analysis: Prefixes and Suffixes** ● Use programming to explain meanings of prefixes and suffixes. **Animating Poetry** ● Create an animated reading of a poem with literal and metaphorical meaning. **Nonfiction Animated Recordings**

	● Use events to create an animated reading of a nonfiction text. **Punctuation Game** ● Use if/then/else conditionals to add punctuation to dialogue. **Creative Storytelling** ● Plan and animate a story using events and sequences.

Coding in Social Studies

Students use programming to model environmental adaptation, explore culture, and celebrate global traditions.

Objectives / Topics Covered	● Animating how communities adapt to environments ● Designing interactive cultural activities ● Representing global celebrations
Lessons	**Communities Adapt to & Modify Their Environment** ● Create a scene showing community adaptation to environments. **Choose Your Own Path: Elements of Culture** ● Create a cultural choose-your-own-path game. **Winter Celebrations Around the World** ● Create a program to tell how a specific holiday is celebrated. **Impacts of Computing: Introduction** ● Create a project comparing past and present technology and its impacts.

Cross-Curricular

Coding lessons that can be integrated into any subject area.

Objectives / Topics Covered	● Integrating coding into multiple subjects ● Using broadcast messages and interactive designs
Lessons	**Digital Flashcards** ● Use broadcast messages to build digital flashcards. **Digital Venn Diagram** ● Program an interactive Venn diagram to compare two things. **Cause and Effect** ● Create a program demonstrating cause-and-effect relationships.

More Coding Lessons & Projects

Students build skills through creative projects, custom art, advanced programming concepts, and seasonal activities.

Objectives / Topics Covered	● Creating custom sprites and backdrops ● Using broadcast messages, conditionals, and variables ● Designing greeting cards and study tools
Lessons	**Costumes, Backdrops, and Animations** ● Create a program with animated sprites and backdrops. **Drawing Tools** ● Create customized sprites and backdrops. **Broadcast Messages: Marco Polo** ● Use broadcast messages to trigger action between sprites. **Introduction to Conditionals**

	• Explain and use if/then blocks in a program. **Variables** • Explain and use variables in a program. **Digital Greeting Card** • Create a digital greeting card using loops and events.

Interdisciplinary Computer Science Third 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.
These resources and more are found on the **CodeHop Resources Page**.	