1st Grade Computer Science Course Syllabus
One Year for Elementary School, 36 Hours

Course Overview and Goals

The **1st Grade Computer Science Course** introduces students to foundational programming concepts through **ScratchJr**, a block-based programming language. Students will develop computational thinking and problem-solving skills while learning to create interactive projects, animations, and games. This course emphasizes creativity and collaboration, providing students with a solid base in computer science concepts and digital literacy.

Learning Environment: This course is designed to be teacher-led, with ready-to-use lesson plans that follow a structured format: **Introduction, Guided Practice, Independent Practice, Extension, and Reflection**. Lessons are built with spiral review to reinforce key concepts and culminate in engaging projects to showcase student understanding.

The lessons are delivered in an **"I do, we do, you do"** format, ensuring a gradual release of responsibility and fostering confidence in students as they learn. Teachers can adapt the content to fit their schedule and instructional needs. The concepts taught in this course spiral across grade levels, ensuring that students can revisit and build upon their understanding year after year, even if all lessons are not completed within a single year. The course includes a total of **36 lessons**, each approximately 45 minutes long. This provides a full school year of material if teaching one lesson per week. Optional digital literacy lessons are also available to complement the programming curriculum with non-programming computer and technology skills.

Programming Environment: Students will write and run programs in **ScratchJr** embedded and saved in students' accounts. The environment supports interactive, hands-on programming, enabling students to create and debug projects in a user-friendly interface.

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/20683/overview?lang=en

A clickable PDF can be found at https://codehs.com/K5CSRoadmaps.

Course Breakdown

Unit 1: Optional Review (7 weeks)

In this unit, students review key computing and ScratchJr skills by following a story-driven sequence of lessons with Scout, practicing sequencing, motion, events, and computational thinking through interactive projects.

Objectives / Topics Covered	● Log in and navigate the CodeHop Playground. ● Use the ScratchJr interface to create characters, scenes, and sequences. ● Apply motion blocks, events, and storytelling through scaffolded animation projects. ● Use computational thinking to break down and sequence real-life routines.
Lessons	**Welcome to CodeHop! (15 minute lesson)** ● Log in and explore the CodeHop Playground as an introduction or warm-up before full lessons. **Scout Adventures 1: Introducing Scout** ● Explore the ScratchJr interface and add Scout or other characters to the stage. **Scout Adventures 2: Scout Starts Exploring** ● Add backgrounds and new pages to create a multi-scene ScratchJr project. **Scout Adventures 3: Scout Meets a Friend** ● Modify and delete characters to develop the story and customize the scene. **Scout Adventures 4: Scout Explores the Forest** ● Use motion blocks to animate Scout moving around the ScratchJr stage. **Scout Adventures 5: Scout and Bluebird Help** ● Build a sequence of motion blocks to move characters around the stage and collect objects. **Scout Adventures 6: Scout Celebrates with Friends** ● Create a celebration scene using characters, pages, backgrounds, motion blocks, and events. **Computational Thinking: Evening Routines** ● Break down evening routines using computational thinking strategies like pattern recognition, sequencing, and simplification.

Unit 2: Sequences and Events (9 weeks)

In this unit, students build on their understanding of sequences and events by creating interactive scenes, animations, and simple games while also learning to debug and use new motion and timing blocks in ScratchJr.

Objectives / Topics Covered	● Use sequences and events to animate characters and tell interactive stories. ● Apply visual effect blocks like hide, grow/shrink, and wait to enhance animations. ● Practice debugging by finding and correcting errors in code. ● Explore creative tools like drawing to customize scenes.
Lessons	**Drawing Tools: Nature Walk** ● Use ScratchJr's drawing tools to design a custom nature-themed scene. **Events** ● Explain what an event is in programming and use multiple event blocks to trigger character actions. **Sequences: Digital Responsibilities** ● Use sequences to program characters and illustrate responsible online behavior. **Hide and Seek Game** ● Create an interactive hide-and-seek game using the hide block to control character visibility. **Grow and Shrink Blocks in Motion** ● Animate characters by combining motion blocks with grow and shrink blocks to change their size dynamically. **Introduction to the Wait Block** ● Use the wait block to add timed pauses in a program, helping control pacing and

timing of actions.
Introduction to Debugging
- Identify and fix bugs in ScratchJr sequences to make programs run as intended.

Careers in CS: Coding for Fashion-Retail
- Explain how coding helps create and improve fashion designs and will create a program to design and animate a fashion character.

Practicing the Design Process
- Use the design process to plan, create, and revise a ScratchJr project that models a solution to a simple real-world problem.

Unit 3: Loops (7 weeks)

Students explore how loops can make code more efficient by repeating actions, applying both repeat and forever loops in unplugged and ScratchJr-based activities.

Objectives / Topics Covered	- Create and follow sequences that use loops to repeat instructions. - Use repeat and forever loops in both unplugged games and ScratchJr projects. - Combine loops with motion, visibility, and events to build interactive animations.
Lessons	**Coding Card Game: Loops** ● Work in groups to guide Scout through a maze using a sequence of instructions that includes loops. **Coding Card Game: Loops 2** ● Continue practicing loop-based logic in a second maze activity using coding cards. **Introduction to Repeat Loops** ● Use repeat loops in ScratchJr to run a section of code multiple times efficiently. **Loops: Catching Butterflies** ● Create an interactive game using show/hide blocks and repeat loops to simulate catching butterflies. **Loops: Predator and Prey (2-part lesson)** ● Use events, sequences, and loops to animate interactions between two animal characters. **Forever Loop Dance Party** ● Create an ongoing dance animation using forever loops to repeat character movements continuously.

Unit 4: Message Events (7 weeks)

In this unit, students use message events in ScratchJr to coordinate interactions between characters, build animated scenes, and explore debugging, timing, and storytelling through creative programming projects.

Objectives / Topics Covered	- Use message events to trigger interactions between characters. - Apply visual and timing blocks with messages to animate dynamic scenes. - Identify and fix bugs in ScratchJr programs. - Create original stories using sequences, events, and visual effects.
Lessons	**Introduction to Message Events** ● Program a relay race where characters pass messages to trigger sequential actions. **Message Events: Simon Says** ● Use message events to make one character send commands to multiple characters in a "Simon Says" game. **Debugging** ● Describe what bugs are and debug a ScratchJr program by finding and correcting errors in sequences. **Create an Original Story Animation** ● Design and animate an original story using characters, backgrounds, sequences, and message events.

	Speed Block: Bouncy Ball
	● Use speed blocks and messages to animate a ball that moves at different speeds.
	Garden Project (2-part lesson)
	● Animate seeds growing into plants by combining message events with grow, shrink, hide, and show blocks.

Unit 5: Pages (4 weeks)

In this unit, students combine multiple ScratchJr concepts—pages, loops, events, and messages—to build interactive projects that demonstrate creativity, storytelling, and real-world impact.

Objectives / Topics Covered	● Use the "go to page" block to navigate between scenes in interactive projects. ● Combine messages, loops, and events to create games and animations. ● Communicate ideas about technology's role in the world through coding. ● Create interactive digital products with expressive and functional elements.
Lessons	**Pages: Create a Tapping Game** ● Build a tapping game that moves between pages using the "go to page" block for game progression. **Create a Mini Golf Game** ● Program a mini golf game using message events and loops to simulate aiming and movement. **Impacts of Technology in Our World** ● Create a ScratchJr animation that demonstrates how technology affects people and communities. **Digital Greeting Card** ● Design a digital card using events and loops to animate a celebratory or seasonal message.

Unit 6: Grid (4 weeks)

In this unit, students explore how to use the grid and end blocks in ScratchJr to control movement, complete projects like mazes and races, and create structured animations.

Objectives / Topics Covered	● Use the ScratchJr grid to plan and program precise character movement. ● Design interactive mazes and animations using grid-based navigation. ● Use the "end" block to signal completion in animations or games.
Lessons	**Grid: Solving Mazes** ● Design a maze and use the grid to program a character to move step-by-step through it. **End Block: Program a Race** ● Create a racing animation and use the "end" block to indicate when a character finishes the race. **Grid: Arctic Animation (2-part lesson)** ● Use the ScratchJr grid to create a winter-themed animation with characters moving across a snowy scene.

Unit 7: Culmination Projects (5 weeks)

In this unit, students demonstrate their understanding of ScratchJr programming concepts by designing original projects that highlight their creativity, storytelling, and game-building skills.

Objectives / Topics Covered	● Plan and create personal or story-based projects using ScratchJr. ● Apply sequences, loops, events, messages, pages, and speed blocks. ● Adjust game mechanics like difficulty and animation flow using code.
Lessons	**About Me Project (3-part lesson)**

	● Plan and build a program that shares personal characteristics and interests using animation and narration. **River Crossing Game (2-part lesson)** ● Create an interactive game with moving obstacles and control difficulty using speed blocks.

Unit 8: Digital Literacy (8 weeks)

In this supplemental unit, students build foundational digital literacy by learning about computers, safe and responsible technology use, data collection, attribution, and how artificial intelligence recognizes patterns.

Objectives / Topics Covered	● Identify computer parts and understand how we interact with technology. ● Practice responsible digital citizenship and recognize personal vs. private information. ● Collect data and present it visually through simple programs. ● Explore how AI and machine learning work by recognizing patterns and attributes.
Lessons	**Computer Basics: Exploration** ● Learn what a computer is, how to use it, and how to identify input, output, hardware, and software. **Keeping Information Safe** ● Identify private and personal information and discuss how to protect it online. **Responsible Digital Citizens** ● Understand digital citizenship by exploring digital footprints, cyberbullying, and how to report online concerns. **Basic Data and Programming** ● Collect basic data and create a simple program to represent the results visually. **Giving Credit Through Attributions** ● Explore the importance of giving credit to original creators and practice attribution by remixing part of a story. **Guided Research (2-part lesson)** ● Use guided research to find information and present findings visually in a ScratchJr program. **Machine Learning: What is a Blorg?** ● Learn how AI recognizes patterns in data and understand how machines distinguish humans through recognition software.

K-1 Course Supplemental Materials

Resources	Description
Parent Welcome Letter (Spanish)	Send this letter home to introduce families to their CS curriculum.
Warm-Up Activities	This warm-up slide deck provides 5-10 minute problems aligned with CS 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 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**.	