



South Carolina Kindergarten Computer Science and Digital Literacy Course Syllabus

One Year for Elementary School, 18 Hours

Course Overview and Goals

The **South Carolina Kindergarten Computer Science and Digital Literacy Course** introduces students to foundational programming concepts through **CodeHop**, a block-based programming language. Students explore digital literacy and computer science skills while engaging in lessons that integrate coding into math, science, English language arts (ELA), and social studies. This course emphasizes creativity, collaboration, and real-world connections, providing students with a strong foundation in both academic content, digital literacy, and computer science.

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 **18 standards-aligned lessons** and **19 exploration lessons**, each approximately 30-45 minutes long. This provides a full school year of material if teaching one lesson per week.

Programming Environment: Students will write and run programs in **CodeHop** 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/27364/overview>



A clickable PDF can be found at <https://codehs.com/SCRoadmap>

Course Breakdown

Unit 1: Getting Started (5 lessons)

This module introduces students to basic computer usage skills—including logging in, mouse and keyboard operations, computational thinking, and computer components—to build confidence using technology independently.

Objectives / Topics Covered	• Basic computer use and logging in • Mouse and keyboard skills • Computational thinking (patterns, sequencing) • Computer parts, hardware/software, input/output • Simple troubleshooting
Lessons	Welcome to CodeHop! • Intro to logging in and navigating the CodeHop Playground. Mouse Practice • Practice clicking, dragging, and basic mouse control through interactive games. Keyboard Introduction • Learn the letters, numbers, and basic functions of the keyboard. Computational Thinking: Morning Routines • Use patterns, sequencing, and task breakdown to model a morning routine. Computer Basics: Introduction • Explore what a computer is, its parts (hardware/software, input/output), and simple troubleshooting steps.

Unit 2: Getting Started: Programming (7 lessons)

This module introduces students to programming by teaching them to use drawing tools, add and modify characters and backgrounds, and create simple motion-based sequences through a story-driven adventure.

Objectives / Topics Covered	• Use drawing tools • Add, edit, and delete characters • Add backgrounds and pages • Use motion blocks • Build simple sequences • Create scenes with events and actions
Lessons	Scout Adventures 1: Introducing Scout • Explore the CodeHop interface and add characters. Scout Adventures 2: Scout Starts Exploring • Add backgrounds and an additional page to expand the scene. Scout Adventures 3: Scout Meets a Friend • Modify and delete characters to adjust the story. Scout Adventures 4: Scout Explores the Forest • Use motion blocks to move characters around the stage. Scout Adventures 5: Scout and Bluebird Help • Build sequences of motion blocks to navigate and collect objects. Scout Adventures 6: Scout Celebrates with Friends • Create a celebration scene using characters, pages, backgrounds, and motion sequences with events. Drawing Tools: Fairy Tale Painting • Use drawing tools to create a custom fairy-tale scene.

Unit 3: Digital Literacy (6 lessons)

This module teaches students how technology connects people, how to choose appropriate software, how to use technology safely, and how data is stored, protected, and interpreted.

Objectives / Topics Covered	• How networks connect people and devices • Types of software and their uses • Safe and responsible device use • Private vs. personal information • How files store data • Collecting and analyzing simple data
Lessons	Using Networks to Connect • Explain how people and devices share information over wired and wireless networks. Types of Software • Compare different kinds of software and select tools for various tasks. Introduction to Responsible Technology Use • Learn safe, respectful technology use and understand Acceptable Use Policies. Keeping Information Safe • Identify private and personal information and how to protect it. Introduction to Data Storage and Files • Recognize that computers store data in files and model simple data collection and storage. What Can Data Tell Us? • Collect and interpret data related to school transportation.

Unit 4: Programming Skill Development (10 lessons)

This module deepens students' programming skills by introducing sequences, events, character behaviors, multi-page programs, sound and speed blocks, and foundational debugging strategies.

Objectives / Topics Covered	• Understanding events and sequences • Creating step-by-step instructions • Using events, pages, sound, and size blocks • Storytelling through code • Basic debugging
Lessons	Acting with Events • Act out how events trigger actions in programs. Coding Card Game: Sequences • Create sequences of instructions to guide Scout through a maze. Sequences • Develop step-by-step instructions through a dance activity. Introduction to Events • Use different types of events to start actions in programs. Introduction to Grow and Shrink Blocks • Use size-change blocks to grow and shrink characters. Introduction to Pages • Create programs using multiple pages. Transportation Speeds and Sounds • Use sound, speech, and speed blocks to animate transportation scenes. All About Me! (2 classes) • Make a program that shares personal favorites through animation. Debugging: Events and Motion • Identify and fix errors in sequences and event-based programs.

Unit 5: Interdisciplinary Exploration (9 lessons)

This module integrates programming with subjects like economics, responsibility, phonics, science, math, and art, enabling students to model real-world concepts using events, messages, and interactive programs.

Objectives / Topics Covered	• Using events/messages to model concepts • Programming characters to explain ideas • Creating phonics and storytelling projects • Modeling science concepts • Coding math story problems and number models • Creating and combining interactive shapes
Lessons	Economic Choices • Use message events to model decision-making and choices between wants and needs • Standard: K.E.2 Our Responsibilities • Program characters to explain responsibility at home and school. • Standard: K.CG.2 Phonics: Letter Sounds • Create a phonics game using tap events and recorded audio. • Standard: ELA.K.F.3.3 Draw and Tell • Create a scene and write or narrate a sentence about it. • Standard: SC/CCSS: K.W.2 Needs of Plants • Decompose and program events to model what plants need to survive. • Standard: K-LS1-1 How Living Things Survive • Illustrate and explain survival strategies of living things. • Standard: K-ESS2-2 Story Problems: Add and Subtract within 10 • Represent addition or subtraction story problems through animated scenes. • Standard: K.OA.1, K.OA.2 Decompose Numbers Up to 10 • Build an interactive program visualizing number decomposition. • Standard: K.OA.3 Creating Shapes • Use events to animate shapes and combine them into a more complex design. • Standard: K.G.1, K.G.2

Kindergarten 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 .	