New York Computer Science and Digital Fluency: 1st Grade Course Syllabus

One Year for Elementary School, 36 Hours

Course Overview and Goals

The **New York Computer Science and Digital Fluency: 1st Grade** introduces students to foundational programming concepts through 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.

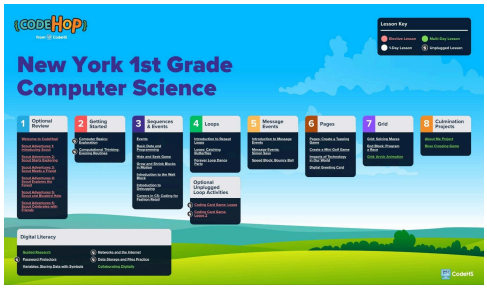
Learning Environment: This course is teacher-led and includes ready-to-use lessons following a consistent structure: **Introduction, Guided Practice, Independent Practice, Extension, and Reflection**. Instruction follows an "I do, we do, you do" model and incorporates spiral review to reinforce concepts and build confidence over time.

The course includes **36 lessons**, each approximately **45 minutes** long, providing a full year of instruction when taught once per week. While the course allows for instructional flexibility, some lessons are required to fully meet state computer science standards and are clearly identified within the syllabus. Required lessons are labeled with the specific standards they address to support planning and compliance.

Programming Environment: Students will write and run programs that are 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/NY_1/overview



A clickable PDF can be found at https://codehs.com/NY-K-5Roadmaps

Course Breakdown

Optional Review

Students are introduced to the CodeHopJr programming environment through the Scout Adventures story-driven series, exploring how to add characters, backgrounds, motion blocks, and sequences.

Objectives / Topics Covered	● Log in to and navigate the CodeHopJr Playground ● Add, modify, and delete characters and backgrounds ● Use motion blocks to move characters on the stage ● Build sequences of blocks and trigger actions with events
Lessons	**Welcome to CodeHop!** ● Log in and explore the CodeHopJr Playground interface for the first time. **Scout Adventures 1: Introducing Scout** ● Explore the programming interface and add characters to a new program. **Scout Adventures 2: Scout Starts Exploring** ● Add backgrounds and a new page to a program. **Scout Adventures 3: Scout Meets a Friend** ● Delete and modify characters already placed in a program. **Scout Adventures 4: Scout Explores the Forest (K-1.CT.4)** ● Use motion blocks to move a character around the stage. **Scout Adventures 5: Scout and Bluebird Help (K-1.CT.4)** ● Build a sequence of motion blocks to move characters and collect objects. **Scout Adventures 6: Scout Celebrates with Friends (K-1.CT.4)** ● Create a celebration scene combining characters, pages, backgrounds, and motion sequences triggered by events.

Unit 1: Getting Started (3 lessons)

Students explore computer fundamentals including hardware and software identification, responsible and safe technology use, and foundational computational thinking skills.

Objectives / Topics Covered	● Identify input, output, hardware, and software ● Explain safe and responsible technology use ● Apply computational thinking to sequence real-world tasks ● Recognize patterns and simplify processes
Lessons	**Computer Basics: Exploration (K-1.IC.6, K-1.IC.7, K-1.NSD.1, K-1.NSD.2, K-1.NSD.3)** ● Identify what a computer is and distinguish input, output, hardware, and software. **Exploring Responsible Technology Use (K-1.CY.5, K-1.DL.7, K-1.IC.2)** ● Explain strategies for using technology safely and responsibly. **Computational Thinking: Evening Routines (K-1.CT.1)** ● Apply computational thinking to identify patterns, sequence steps, and simplify an everyday routine.

Unit 2: Sequences and Events (6 lessons)

Students learn to use events, motion blocks, and data to build interactive programs, while also practicing debugging and exploring computer science careers.

Objectives / Topics Covered	● Use event blocks to trigger actions in a program ● Collect data and represent it visually in a program ● Use hide/show, grow/shrink, and motion blocks together

	● Find and fix bugs in a sequence of code
Lessons	**Events (K-1.CT.4, K-1.CT.6, K-1.CT.9)** ● Explain what an event is and use multiple event blocks to trigger actions. **Basic Data and Programming (K-1.CT.2, K-1.CT.3, K-1.CT.4, K-1.CT.9, K-1.DL.4)** ● Collect class data and build a program that displays it visually. **Hide and Seek Game (K-1.DL.4)** ● Program an interactive hide-and-seek game using the hide block. **Grow and Shrink Blocks in Motion (K-1.CT.6)** ● Create a program that combines motion blocks with grow and shrink blocks to resize characters. **Introduction to Debugging (K-1.CT.4, K-1.CT.8, K-1.CT.9)** ● Locate and correct errors in a sequence of code. **Careers in CS: Fashion-Retail (K-1.CT.4, K-1.CT.6, K-1.DL.4, K-1.IC.7)** ● Describe how coding supports fashion design and animate a fashion character.

Optional Unplugged Loops Activities

Students work together in an unplugged card game to practice writing sequences with loops, guiding Scout through a maze without a computer.

Objectives / Topics Covered	● Collaborate to plan a sequence of instructions ● Use loops to simplify repeated steps in a path ● Debug and revise instructions to complete a maze
Lessons	**Coding Card Game: Loops (K-1.CT.10, K-1.CT.6, K-1.CT.8)** ● Work as a team to build loop-based instructions that move Scout through a maze. **Coding Card Game: Loops 2 (K-1.CT.10, K-1.CT.6, K-1.CT.8)** ● Continue practicing loop sequences to navigate Scout through a more complex maze.

Unit 3: Loops (3 lessons)

Students use repeat and forever loops to run code multiple times, creating games and animations that demonstrate how loops make programs more efficient.

Objectives / Topics Covered	● Use repeat loops to run a block of code multiple times ● Apply show and hide blocks inside a loop ● Use a forever loop to create continuous animations ● Distinguish between repeat and forever loop behaviors
Lessons	**Introduction to Repeat Loops (K-1.CT.4, K-1.CT.6, K-1.CT.8, K-1.CT.9)** ● Use a repeat loop to run a section of code a specified number of times. **Loops: Catching Butterflies (K-1.CT.6, K-1.CT.8, K-1.DL.4)** ● Combine show, hide, and loops to build a butterfly-catching game. **Forever Loop Dance Party (K-1.CT.6, K-1.CT.8)** ● Create an animation using a repeat-forever loop so characters perform actions continuously.

Unit 4: Message Events (3 lessons)

Students program characters to communicate with each other using message events and control character speed, building more interactive multi-character programs.

Objectives / Topics Covered	● Use message events to make characters interact ● Send a message from one character to many characters ● Control character speed using the speed block ● Combine messages and speed blocks in a single program
Lessons	**Introduction to Message Events (K-1.CT.4, K-1.CT.6)** ● Program a relay race in which messages cause characters to interact with each other. **Message Events: Simon Says (K-1.CT.4, K-1.CT.6)** ● Use message events so one character communicates commands to multiple characters simultaneously. **Speed Block: Bouncy Ball (K-1.CT.6, K-1.CT.9, K-1.DL.4)** ● Program a character to move at varying speeds using speed blocks and messages.

Unit 5: Pages (3 lessons)

Students use page navigation blocks along with loops and events to create multi-page interactive projects such as games and digital greeting cards.

Objectives / Topics Covered	● Use go-to-page blocks to move between pages ● Combine messages and loops in a multi-page program ● Plan and build a multi-page interactive project ● Apply events and loops in a creative digital product
Lessons	**Pages: Create a Tapping Game (K-1.CT.6, K-1.DL.4)** ● Build a game that advances to the next page when a character is tapped. **Create a Mini Golf Game (K-1.CT.6, K-1.DL.4)** ● Use messages and loops to design and program a mini golf game. **Digital Greeting Card (K-1.CT.10, K-1.CT.6, K-1.DL.4)** ● Design a digital greeting card that uses events and loops to bring the card to life.

Unit 6: Grid (4 lessons)

Students use the CodeHopJr grid to place characters precisely, design mazes, program races, and create grid-based animations.

Objectives / Topics Covered	● Use the grid to position characters on the stage ● Design and solve a maze using directional programming ● Use the end block to stop an animation or program ● Create an animation using grid-based movement
Lessons	**Grid: Solving Mazes (K-1.CT.4, K-1.CT.6, K-1.CT.9)** ● Design a maze and use the grid to program a character to navigate through it. **End Block: Program a Race (K-1.CT.6, K-1.CT.9, K-1.DL.4)** ● Program a race and use the end block to stop the animation when the race is complete. **Grid: Arctic Animation (2 classes | K-1.CT.4, K-1.CT.6, K-1.DL.4)** ● Use the grid to create a moving arctic-themed animation.

Unit 7: Culmination Projects (5 lessons)

Students apply everything they have learned to design and build two original projects: a personal About Me program and a River Crossing game with adjustable difficulty.

Objectives / Topics Covered	• Plan and build a program that expresses personal interests • Program obstacles and control game difficulty with speed blocks • Apply sequences, loops, events, and messages in a complete project • Reflect on and share finished programs with others
Lessons	**About Me Project (3 classes | K-1.CT.10, K-1.CT.4, K-1.CT.6, K-1.DL.4)** • Plan and create a program that describes personal characteristics and interests. **River Crossing Game (2 classes | K-1.CT.6, K-1.CT.9, K-1.DL.4)** • Program a river crossing game with obstacles and adjust difficulty using speed blocks.

Unit 8: Digital Literacy (9 lessons)

Students explore key digital citizenship and computer science concepts including research, password safety, computing impacts, variables, networks, data storage, and digital collaboration.

Objectives / Topics Covered	• Find and present research using digital sources • Protect login information with strong password strategies • Explain how computing devices affect daily life • Model how networks, data storage, and variables work
Lessons	**Guided Research (2 classes | K-1.CT.3, K-1.CT.6, K-1.DL.3, K-1.DL.4, K-1.DL.7)** • Find information from research sources and create a program to share findings visually. **Password Protectors (K-1.CY.1, K-1.CY.2, K-1.CY.4)** • Demonstrate strategies for keeping usernames and passwords safe. **Impacts of Computing: Our Community (K-1.IC.1, K-1.IC.3, K-1.IC.7)** • Identify computing devices in daily life and describe how technology has changed how people live, work, and play. **Variables: Storing Data with Symbols (K-1.CT.7)** • Model how programs store and change data using variables. **Networks and the Internet (K-1.NSD.4)** • Explain what a network is and model how messages travel across the Internet. **Data Storage and Files Practice (K-1.NSD.5)** • Recognize that computers store data as files and model how data is collected and saved. **Collaborating Digitally (2 classes | K-1.DL.2)** • Work with others digitally to create a shared program about the classroom.

New York Computer Science and Digital Fluency: 1st Grade 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**.	