



New York Computer Science and Digital Fluency: 2nd Grade Course Syllabus

One Year for Elementary School, 36 Hours

Course Overview and Goals

The **New York Computer Science and Digital Fluency: 2nd 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.

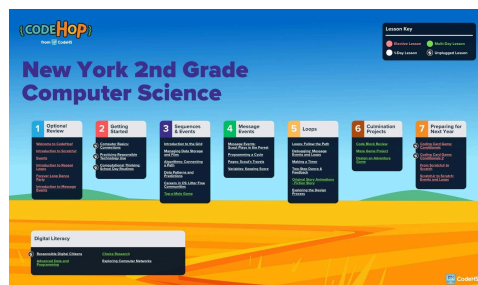
Learning Environment: This course is teacher-led and includes ready-to-use lessons following a consistent structure: **Introduction, Guided Practice, Independent Practice, Extension, and Reflection**. Instruction follows an “I do, we do, you do” model and incorporates spiral review to reinforce concepts 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_2/overview



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

Course Breakdown

Optional Review

Students review core programming concepts from 1st grade — logging in, creating scenes with characters, using events, loops, and message events — to build confidence before starting new material.

Objectives / Topics Covered	<ul style="list-style-type: none">● Log in and navigate the CodeHopJr Playground● Use event blocks to trigger character actions● Apply repeat and forever loops in a program● Use message events to make characters interact
Lessons	<p>Welcome to CodeHop!</p> <ul style="list-style-type: none">● Log in and explore the CodeHopJr Playground interface. <p>Introduction to Programming</p> <ul style="list-style-type: none">● Navigate the programming interface and build a scene with characters. <p>Events (2-3.CT.9)</p> <ul style="list-style-type: none">● Explain what an event is and use multiple event blocks to trigger actions in a program. <p>Introduction to Repeat Loops (2-3.CT.9)</p> <ul style="list-style-type: none">● Use a repeat loop to run a section of code a set number of times. <p>Forever Loop Dance Party (2-3.CT.9)</p> <ul style="list-style-type: none">● Create an animation where characters repeat actions continuously using a forever loop. <p>Introduction to Message Events</p> <ul style="list-style-type: none">● Program a relay race in which messages cause characters to interact.

Unit 1: Getting Started (3 lessons)

Students build foundational knowledge of how computers work, how to use technology responsibly, and how computational thinking applies to everyday routines.

Objectives / Topics Covered	<ul style="list-style-type: none">● Identify and explain input, output, hardware, and software● Demonstrate safe and responsible technology use● Apply computational thinking to sequence and simplify tasks● Recognize patterns and break down problems into steps
Lessons	<p>Computer Basics: Connections (2-3.CY.3, 2-3.IC.1, 2-3.IC.3, 2-3.IC.5, 2-3.NSD.1, 2-3.NSD.3)</p> <ul style="list-style-type: none">● Identify how input, output, hardware, and software work together in a computer system. <p>Practicing Responsible Technology Use (2-3.DL.7, 2-3.IC.2)</p> <ul style="list-style-type: none">● Demonstrate strategies for using technology safely and responsibly. <p>Computational Thinking: School Day Routines (2-3.CT.1)</p> <ul style="list-style-type: none">● Use computational thinking to identify patterns, sequence steps, and simplify a school day routine.

Unit 2: Sequences and Events (7 lessons)

Students use the grid, algorithms, and events to move characters precisely and build interactive programs, while also exploring data patterns and computer science careers.

Objectives / Topics Covered	<ul style="list-style-type: none">● Use the grid to position characters on the stage● Create and adjust algorithms to move characters● Identify patterns and trends in data visualizations● Build interactive programs using events
-----------------------------	--

Lessons	<p>Introduction to the Grid</p> <ul style="list-style-type: none"> Use the grid to move characters to specific locations on the stage. <p>Managing Data Storage and Files (2-3.NSD.5)</p> <ul style="list-style-type: none"> Recognize that computers store data as files and model how data is collected and stored. <p>Algorithms: Connecting a Path (2-3.CT.4, 2-3.CT.6, 2-3.CT.9, 2-3.DL.4)</p> <ul style="list-style-type: none"> Create and adjust algorithms that move characters based on their size, shape, and starting position. <p>Data Patterns and Predictions (2-3.CT.3)</p> <ul style="list-style-type: none"> Identify trends in data visualizations and create a program that communicates patterns and predictions. <p>Careers in CS: Litter Free Communities (2-3.IC.3, 2-3.IC.7)</p> <ul style="list-style-type: none"> Explain how computer science helps solve community problems and build a litter-sorting program. <p>Tap-a-Mole Game (2 classes 2-3.CT.10, 2-3.CT.9)</p> <ul style="list-style-type: none"> Create an interactive tap-a-mole game using event blocks.
---------	--

Unit 3: Message Events (4 lessons)

Students deepen their use of message events to control program flow, model cycles, navigate between pages, and track data with variables.

Objectives / Topics Covered	<ul style="list-style-type: none"> Use message events to control the flow of a program Model a real-world cycle using message events Navigate between pages using messages Store and update data using a variable
Lessons	<p>Message Events: Scout Plays in the Forest (2-3.DL.4)</p> <ul style="list-style-type: none"> Use message events to control the order in which actions happen in a program. <p>Programming a Cycle</p> <ul style="list-style-type: none"> Use message events to represent a repeating real-world cycle in a program. <p>Pages: Scout's Travels (2-3.CT.9)</p> <ul style="list-style-type: none"> Use messages to move Scout between pages in a multi-page program. <p>Variables: Keeping Score (2-3.CT.7, 2-3.DL.4, 2-3.NSD.5)</p> <ul style="list-style-type: none"> Create a program that uses a variable to keep and update score.

Unit 4: Loops (7 lessons)

Students use loops to solve pattern-based problems, debug programs, build timers, and complete creative projects including animations and design-process challenges.

Objectives / Topics Covered	<ul style="list-style-type: none"> Identify patterns and apply loops to model them Debug loop and message event errors in code Use loops and wait blocks to build a timer Revise a program based on peer feedback
Lessons	<p>Loops: Follow the Path (2-3.CT.1)</p> <ul style="list-style-type: none"> Identify a pattern and write a loop-based program to follow a path. <p>Debugging: Message Events and Loops (2-3.CT.9)</p> <ul style="list-style-type: none"> Locate and fix errors in programs that use message events and loops. <p>Making a Timer (2-3.CT.9)</p> <ul style="list-style-type: none"> Use loops, wait blocks, and turn blocks to create and compare two timers running at different speeds. <p>Two-Step Dance & Feedback (2-3.CT.9)</p> <ul style="list-style-type: none"> Create a dance program, give and receive peer feedback, and revise the program

	<p>based on that feedback.</p> <p>Original Story Animations - Fiction Story (2 classes 2-3.CT.10)</p> <ul style="list-style-type: none"> • Develop an original fiction story and animate it as a program. <p>Exploring the Design Process</p> <ul style="list-style-type: none"> • Use the design process to plan, build, and improve a loop-based program that models a real-world solution.
--	---

Unit 5: Culmination Projects (8 lessons)

Students synthesize their learning by reviewing coding blocks, building and revising an interactive maze game, and designing a multi-page adventure game from scratch.

Objectives / Topics Covered	<ul style="list-style-type: none"> • Use and explain a variety of coding blocks in a program • Build an interactive game using events, messages, and loops • Revise a program based on peer feedback • Design and create a multi-page story-based game
Lessons	<p>Code Block Review (2 classes 2-3.CT.10, 2-3.CT.9, 2-3.DL.4)</p> <ul style="list-style-type: none"> • Use a variety of coding blocks and explain what each one does in a program. <p>Maze Game Project (3 classes 2-3.CT.9)</p> <ul style="list-style-type: none"> • Build and explore multiple ways to program an interactive maze game, then revise it based on feedback. <p>Design an Adventure Game (3 classes 2-3.CT.10, 2-3.CT.9)</p> <ul style="list-style-type: none"> • Create a story-based, multi-page adventure game using all the CS skills learned in the course.

Unit 6: Digital Literacy (7 lessons)

Students explore digital citizenship, data collection, research skills, and computer networks, connecting programming to real-world contexts of technology use and online responsibility.

Objectives / Topics Covered	<ul style="list-style-type: none"> • Explain how technology has changed how people live and work • Define digital citizenship and describe responsible online behavior • Collect and present data visually using a program • Describe how networks connect devices to share information
Lessons	<p>Impacts of Computing: Changing World (2-3.IC.1)</p> <ul style="list-style-type: none"> • Identify computing devices in daily life and create a program explaining how technology has changed the world. <p>Responsible Digital Citizens (2-3.CY.1, 2-3.CY.2, 2-3.CY.5, 2-3.DL.6, 2-3.DL.7)</p> <ul style="list-style-type: none"> • Explain digital footprints, discuss cyberbullying, and describe how to report concerns online. <p>Data Explorers (2 classes 2-3.CT.2, 2-3.CT.3, 2-3.DL.2, 2-3.NSD.4)</p> <ul style="list-style-type: none"> • Develop a survey question, collect data, and create a program that presents findings visually. <p>Choice Research (2 classes 2-3.CT.3, 2-3.DL.3)</p> <ul style="list-style-type: none"> • Gather and evaluate sources to answer a research question and share findings through a program. <p>Exploring Computer Networks</p> <ul style="list-style-type: none"> • Describe how networks connect devices and model how information is sent and received using message blocks.

Preparing for Next Year

Students preview 3rd grade concepts through unplugged conditional card games and hands-on introductions to the CodeHop block-based editor.

Objectives / Topics Covered	<ul style="list-style-type: none"> • Use conditionals in a sequence of instructions • Navigate the CodeHop editor interface • Create a simple program in CodeHop using events and loops
Lessons	<p>Coding Card Game: Conditionals (2-3.CT.8)</p> <ul style="list-style-type: none"> • Collaborate to build a conditional-based instruction sequence that moves Scout through a maze. <p>Coding Card Game: Conditionals 2 (2-3.CT.8)</p> <ul style="list-style-type: none"> • Continue practicing conditionals to navigate Scout through a more challenging maze. <p>From CodeHopJr to CodeHop Blocks (2-3.DL.4)</p> <ul style="list-style-type: none"> • Navigate the CodeHop editor and create a simple program using its block interface. <p>From CodeHopJr to CodeHop Blocks: Part 2 (2-3.DL.4)</p> <ul style="list-style-type: none"> • Build a CodeHop program that uses an event and a loop.

New York Computer Science and Digital Fluency: 2nd 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 .	