4th Grade Computer Science Course Syllabus

One Year for Elementary School, 36 Hours

Course Overview and Goals

The 4th Grade Computer Science Course 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.

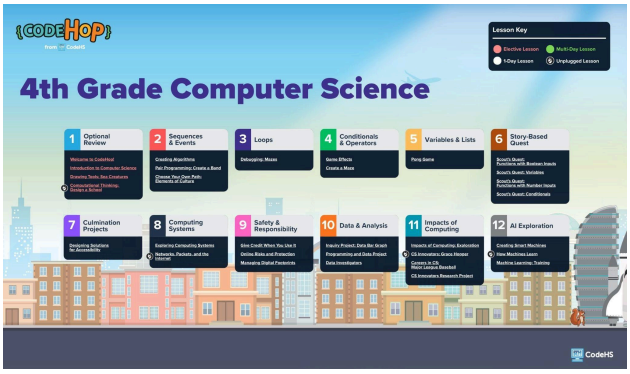
The course includes 36 lessons, each approximately 45 minutes long, providing a full year of instruction when taught once per week. While the course allows for instructional flexibility, some lessons are required to fully meet CSTA computer science standards and are clearly identified within the syllabus.

Standards Alignment Note: Lessons that list standards in parentheses next to the title are required to fully meet CSTA computer science standards. Lessons without a standards tag support spiral review, practice, or enrichment.

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/28416/overview



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

Course Breakdown

Unit 1: Optional Review (4 lessons)

These optional lessons are available for review at the start of 4th grade. They are not counted toward the 36-lesson course total.

Objectives / Topics Covered	● Review foundational CodeHop navigation and programming concepts. ● Revisit sequences, events, and computational thinking from prior grades. ● Build confidence before beginning new 4th grade content.
Lessons	**Welcome to CodeHop!** ● Learn how to log in and use the CodeHop Playground. This short introductory lesson can be used on its own, or right before a full lesson. **Introduction to Computer Science** ● Define important computer science vocabulary and create a simple program. **Drawing Tools: Sea Creatures** ● Use all of the CodeHop image editing tools to create and program deep sea creatures. **Computational Thinking: Design a School** ● Use computational thinking to design a school.

Unit 2: Sequences & Events (3 lessons)

In this module, students apply sequencing and event programming in collaborative and culturally creative contexts, designing algorithms and building a multi-scene interactive story.

Objectives / Topics Covered	● Design and implement algorithms to solve programming challenges. ● Collaborate using pair programming techniques. ● Apply sequencing and events to build a multi-path interactive story.
Lessons	**Creating Algorithms** (E4-ALG-PS-01, E4-PRO-PD-04, E4-ALG-IM-03) ● Define important computer science vocabulary and create a simple program. **Pair Programming: Create a Band** (2 classes | E4-PRO-PD-05, E4-PRO-TR-08) ● Collaborate through pair programming to design and code a band in CodeHop using keyboard inputs. **Choose Your Own Path: Elements of Culture** (2 classes | E4-ALG-PS-01) ● Identify elements of culture as they create a cultural choose-your-own-path game.

Unit 3: Loops (1 lesson)

In this module, students apply debugging skills to programs that use loops in a maze challenge.

Objectives / Topics Covered	● Identify and fix errors in programs that use loops. ● Apply systematic debugging strategies to structured maze programs.
Lessons	**Debugging: Mazes** (E4-PRO-TR-08) ● Decompose a program to debug and make the program run as intended.

Unit 4: Conditionals & Operators (2 lessons)

In this module, students apply conditionals and operators to build interactive game programs.

Objectives / Topics Covered	● Use conditional blocks and Boolean operators to create decision-based programs. ● Design and build a complete interactive game using conditionals and operators.
Lessons	**Game Effects** ● Modify a game to add engaging effects and make updates to their game based on peer feedback. **Create a Maze** (2 classes | E4-PRO-TR-08) ● Draw a maze backdrop in CodeHop and program Scout to navigate through the maze.

Unit 5: Variables & Lists (1 lesson)

In this module, students apply variables and game mechanics to build a complete Pong game.

Objectives / Topics Covered	● Use variables to store and update game data such as score. ● Apply all programming concepts to build a complete Pong game.
Lessons	**Pong Game** (2 classes | E4-PRO-VD-06) ● Create and use variables to keep score in an interactive pong game.

Unit 6: Story-Based Quest (4 lessons)

In this module, students deepen their understanding of functions, variables, and conditionals through Scout's Quest — a four-part story-based programming series.

Objectives / Topics Covered	● Create and call functions that use Boolean inputs. ● Use variables to store and manage data in programs. ● Write functions that accept number inputs to produce different outputs. ● Apply conditionals within functions to create decision-based code.
Lessons	**Scout's Quest: Functions with Boolean Inputs** ● Create a function including a boolean input to perform different actions based on whether a password is correct. Part 1 of 4 in Scout's Quest skill review series. **Scout's Quest: Variables** (E4-PRO-VD-06) ● Create and use variables to track points in a program. Part 2 of 4 in Scout's Quest skill review series. **Scout's Quest: Functions with Number Inputs** ● Create a drawing using functions with number inputs. Part 3 of 4 in Scout's Quest skill review series. **Scout's Quest: Conditionals** ● Create a program using if/then conditionals. Part 4 of 4 in Scout's Quest skill review series.

Unit 7: Culmination Projects (1 lesson)

In this module, students apply the full design process to create an accessibility-focused programming project over multiple class periods.

Objectives / Topics Covered	● Apply the design process to create a program that addresses an accessibility need. ● Collaborate to ideate, prototype, and reflect on a user-centered design solution.
Lessons	**Designing Solutions for Accessibility** (2 classes | E4-PRO-PD-05) ● Use the design thinking process to identify and solve real-world problems by redesigning a game to improve accessibility and usability for diverse users.

Unit 8: Computing Systems (2 lessons)

In this module, students explore how computing systems work and how data travels across networks and the Internet.

Objectives / Topics Covered	● Identify and describe how computing systems are organized and how components interact. ● Explain how data is broken into packets and transmitted across the Internet.
Lessons	**Exploring Computing Systems** (E4-SYS-HW-12) ● Identify parts of the computing system and identify simple hardware and software problems. **Networks, Packets, and the Internet (Unplugged)** (E4-SYS-NT-14) ● Identify parts of the computing system and identify simple hardware and software problems.

Unit 9: Safety & Responsibility (3 lessons)

In this module, students explore digital attribution, online risks, and digital footprint management.

Objectives / Topics Covered	● Apply proper attribution practices when using others' creative work. ● Identify online risks and apply strategies to protect personal information. ● Manage and reflect on a personal digital footprint.
Lessons	**Give Credit When You Use It** (E4-PRO-RD-07) ● Search for information to answer questions online and provide proper attribution to sources. **Online Risks and Protection** (E4-SYS-SE-13) ● Demonstrate how to stay safe online by practicing secure habits and understanding the tools and technologies that protect their information. **Managing Digital Footprints** ● Explain how online actions create permanent digital footprints and describe how to manage their digital identity responsibly.

Unit 10: Data & Analysis (3 lessons)

In this module, students collect, visualize, and analyze data across three projects, connecting data literacy with programming.

Objectives / Topics Covered	● Collect and organize data and create a visual representation of findings. ● Connect data collection and analysis with programming projects. ● Interpret data to answer questions and identify patterns.

Lessons	**Inquiry Project: Data Bar Graph** (2 classes | E4-DAT-DC-09, E4-DAT-DI-10) ● Follow the inquiry process and modify a program to display the results of their investigation. **Programming and Data Project** (2 classes | E4-DAT-DC-09) ● Develop an investigative question, collect data, draw conclusions based on the data, and create an interactive program to present data visually. **Data Investigators** (E4-DAT-DI-10, E4-DAT-IM-11) ● Evaluate data for reliability and then analyze the data to draw conclusions and make predictions.

Unit 11: Impacts of Computing (4 lessons)

In this module, students explore computing's impact on society, learn about CS innovators, and complete a research project.

Objectives / Topics Covered	● Describe how computing has shaped and continues to impact society. ● Research and present information about a significant CS innovator. ● Explore how computer science is applied in professional fields.
Lessons	**Impacts of Computing: Exploration** (E4-SYS-IM-15, E4-SOC-CE-19) ● Evaluate data for reliability and then analyze the data to draw conclusions and make predictions. **CS Innovators: Grace Hopper (Unplugged)** (E4-SOC-HI-16) ● Explain how Grace Hopper's work was important to computer science and use binary code to decompose mystery words. **Careers in CS: Major League Baseball** (E4-SOC-CE-19) ● Explain how coding can be used in sports, and abstract events from an article to retell important events in a timeline program. **CS Innovators Research Project** (2 classes | E4-SOC-HI-16) ● Research an innovator and abstract facts from an article to use as an informational program.

Unit 12: AI Exploration (3 lessons)

In this module, students explore how AI systems are created, how machine learning works, and experience the training process firsthand.

Objectives / Topics Covered	● Explain what makes a machine 'smart' and how AI systems are designed. ● Describe how machine learning models learn from labeled training data. ● Train and evaluate a simple machine learning model.
Lessons	**Creating Smart Machines** (E4-SOC-ET-17) ● Describe how emerging technologies can be used and why people design and build new technology. **How Machines Learn (Unplugged)** (E4-SOC-HU-18) ● Explain the different machine learning approaches and modify a program to model how AI can be trained to make predictions. **Machine Learning: Training** (2 classes | E4-ALG-ML-02) ● Explain the different machine learning approaches and modify a program to model how AI can be trained to make predictions.

4th 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.
Design-Your-Own-Lesson Templates	Empower your students to explore and express their knowledge creatively with our versatile graphic organizer templates. Designed with adaptability and ease of use in mind, these interactive tools transform any subject into an engaging, hands-on learning experience.
These resources and more are found on the **CodeHop Resources Page**.	