Colorado 5th Grade Computer Science Course Syllabus

One Year for Elementary School, 36 Hours

Course Overview and Goals

The **Colorado 5th Grade Computer Science Course** introduces students to foundational programming concepts through 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.

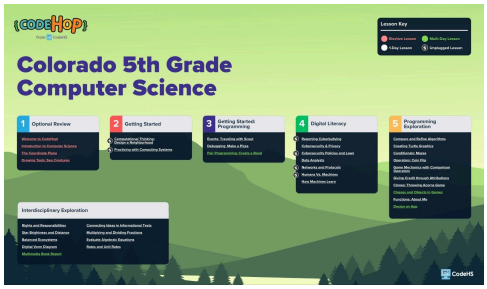
Learning Environment: This course is designed to be teacher-led, with ready-to-use lesson plans. Each programming lesson follows a structured format: **Introduction, Guided Practice, Independent Practice, Extension, and Reflection.** Many digital literacy lessons contain unplugged activities, requiring printed handouts and class activities to support hands-on learning.

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.

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/CO_5/overview



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

Course Breakdown

Optional Review

This optional review unit is designed to support students who would benefit from additional practice before or after completing core lessons.

Objectives / Topics Covered	● Log in and navigate the Playground. ● Create a simple program to demonstrate basic programming skills. ● Use the coordinate plane to position sprites.
Lessons	**Welcome to CodeHop!** ● Learn how to log in and navigate the CodeHop Playground. **Introduction to Computer Science** ● Define key computer science vocabulary and create a simple program. **The Coordinate Plane** ● Create an open-ended animation using coordinate-plane movement. **Drawing Tools: Sea Creatures** ● Use CodeHop imaging tools to create and program deep sea creatures.

Unit 1: Getting Started (2 lessons)

In this introductory unit, students are introduced to the basics of computing by identifying key parts of a computing system and learning how to troubleshoot simple issues. They also begin developing foundational computational thinking skills by designing and sequencing steps to design a neighborhood.

Objectives / Topics Covered	● Identify parts of a computing system. ● Practice basic troubleshooting strategies for common computer problems. ● Apply computational thinking skills.
Lessons	**Computational Thinking: Design a Neighborhood** ● Use computational thinking skills to plan and design a neighborhood. **Practicing with Computing Systems** ● Identify computer components and troubleshoot basic hardware and software issues. ● *Standard Met: CS.5.1.1a*

Unit 2: Getting Started: Programming (4 lessons)

In this unit, students will create basic programs while learning about events, algorithms, and loops. Students will practice debugging their programs to solve simple problems.

Objectives / Topics Covered	● Debug sequences in a program. ● Collaborate through pair programming.
Lessons	**Events: Traveling with Scout** ● Use events to trigger character actions. ● *Standard Met: CS.5.1.2a* **Debugging: Make a Pizza** ● Decompose a program to find and fix bugs so the sequence runs correctly. ● *Standard Met: CS.5.1.1c* **Pair Programming: Create a Band (2 part lesson)** ● Design and code a band using keyboard inputs. ● *Standard Met: CS.5.3.1e*

Unit 3: Digital Literacy (7 lessons)

In this unit, students will learn what it means to be a responsible digital citizen by identifying appropriate behavior and protecting themselves online. Students will also explore digital literacy concepts such as AI, data, and networks.

Objectives / Topics Covered	● Practice safe and healthy habits online. ● Analyze data and make predictions. ● Model how networks function. ● Explain how machines learn.
Lessons	**Reporting Cyberbullying** ● Recognize and compare types of cyberbullying and explain how online interactions can escalate. ● *Standards Met: CS.5.6.1a, CS.5.6.1b* **Cybersecurity & Privacy** ● Demonstrate how to solve a real-world cybersecurity issue. **Cybersecurity Policies and Laws** ● Research school technology policies and examine a cybersecurity law from your state. ● *Standard Met: CS.5.6.1c* **Data Analysts** ● Analyze data to draw conclusions and make predictions. **Networks and Protocols** ● Explore how network protocols enable data transfer and compare WiFi, wired, and cellular networks. **Humans Vs. Machines** ● Compare human and computer performance on similar tasks. ● *Standard Met: CS.5.5.1a* **How Machines Learn** ● Explain different machine learning approaches. ● *Standards Met: CS.5.5.1b, CS.5.5.1c*

Unit 4: Programming Exploration (13 lessons)

In this unit, students will gain basic programming skills using a block-based programming language. Students will practice with sequences, loops, variables, and functions.

Objectives / Topics Covered	● Use loops to simplify code. ● Use conditionals to build interactive and reactive programs. ● Learn how variables store information. ● Use functions to organize code into reusable parts.
Lessons	**Compare and Refine Algorithms** ● Compare and refine multiple algorithms and determine which is most efficient. ● *Standard Met: CS.5.1.1b* **Creating Turtle Graphics** ● Use the pen tool and loops to create turtle-style graphics. ● *Standards Met: CS.5.1.2a, CS.5.3.1b* **Conditionals: Mazes** ● Create a maze program that uses conditionals to guide actions. **Operators: Coin Flip** ● Build a coin-flipping simulation using variables and operators. ● *Standards Met: CS.5.1.2a, CS.5.3.1a, CS.5.3.1b* **Game Mechanics with Comparison Operators** ● Use comparison operators and variables to create game mechanics. ● *Standard Met: CS.5.3.1a* **Giving Credit Through Attributions** ● Give appropriate attribution when creating or remixing programs.

	● *Standard Met: CS.5.3.1c* **Clones: Throwing Acorns Game** ● Create a game using clones. **Classes and Objects in Games (2 part lesson)** ● Use randomizers to change characteristics of objects in a game. ● *Standards Met: CS.5.3.1d, CS.5.3.1f* **Functions: About Me** ● Create and use a function with input to design an interactive “About Me” program. **Design an App (3 part lesson)** ● Use the design thinking process to create an app that solves a user's needs. ● *Standards Met: CS.5.2.1a, CS.5.3.1a, CS.5.3.1b, CS.5.3.1d, CS.5.3.1f*

Unit 5: Interdisciplinary Exploration (10 lessons)

This unit integrates programming with subjects like social studies, math, science, and ELA, enabling students to model real-world concepts using events, algorithms, and interactive programs.

Objectives / Topics Covered	● Use algorithms, loops, and conditionals to model concepts in social studies, math, science, and ELA.
Lessons	**Rights and Responsibilities** ● Create a voting simulation that uses variables and events to model civic concepts. **Star Brightness and Distance** ● Demonstrate how distance affects star brightness using conditionals. **Balanced Ecosystems** ● Build an ecosystem simulation and adjust elements to show how populations change. **Digital Venn Diagram** ● Program an interactive Venn diagram to compare two things. **Multimedia Book Report (2 part lesson)** ● Create a program that communicates key information about a book. **Identifying Main Ideas in Informational Texts** ● Identify the main idea of an informational text and explain 2–3 supporting details that show how the author develops that idea. **Multiplying and Dividing Fractions** ● Model multiplying and dividing fractions through a program that solves word problems. **Evaluate Algebraic Equations** ● Create a program using operators and conditionals to check and solve real-world algebraic equations. **Rates and Unit Rates** ● Design a unit rate converter using variables and operators.

5th 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**.	