Colorado 3rd Grade Computer Science Course Syllabus

One Year for Elementary School, 36 Hours

Course Overview and Goals

The **Colorado 3rd 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.

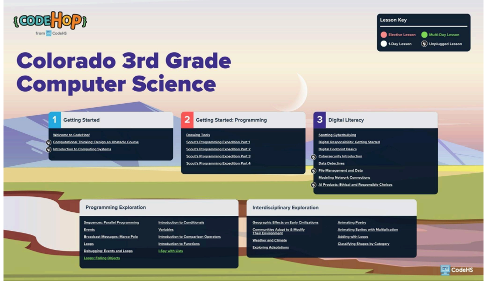
Learning Environment: This course is designed to be teacher-led, with ready-to-use lesson plans. Each programming lesson follows a structured format: **Introduction, Guided Practice, Independent Practice, Extension, and Reflection.** Many digital literacy lessons contain unplugged activities, requiring 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_3/overview



Colorado 3rd Grade Computer Science

Lesson Key

🔴 Elective Lesson
🟢 Multi-Day Lesson
⚪ 1-Day Lesson
Unplugged Lesson

1 Getting Started

- Welcome to CodeHop!
- Computational Thinking: Design an Obstacle Course
- Introduction to Computing Systems

2 Getting Started: Programming

- Drawing Tools
- Scout's Programming Expedition Part 1
- Scout's Programming Expedition Part 2
- Scout's Programming Expedition Part 3
- Scout's Programming Expedition Part 4

3 Digital Literacy

- Spotting Cyberbullying
- Digital Responsibility: Getting Started
- Digital Footprint Basics
- Cybersecurity Introduction
- Data Detectives
- File Management and Data
- Modeling Network Connections
- AI Products: Ethical and Responsible Choices

Programming Exploration

- Sequences: Parallel Programming
- Events
- Broadcast Messages: Marco Polo
- Loops
- Debugging: Events and Loops
- Loops: Falling Objects
- Introduction to Conditionals
- Variables
- Introduction to Comparison Operators
- Introduction to Functions
- I-Spy with Lists

Interdisciplinary Exploration

- Geographic Effects on Early Civilizations
- Communities Adapt to & Modify Their Environment
- Weather and Climate
- Exploring Adaptations
- Animating Poetry
- Animating Sprites with Multiplication
- Adding with Loops
- Classifying Shapes by Category

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

Course Breakdown

Unit 1: Getting Started (3 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 in an obstacle course challenge.

Objectives / Topics Covered	● Log in and navigate the Playground. ● Identify parts of a computing system. ● Practice basic troubleshooting strategies for common computer problems. ● Apply computational thinking skills.
Lessons	**Welcome to CodeHop!** ● Learn how to log in and navigate the CodeHop Playground. **Computational Thinking: Design an Obstacle Course** ● Use computational thinking to design an obstacle course by identifying patterns, sequencing actions, and breaking tasks down. **Introduction to Computing Systems** ● Identify parts of a computer system and solve simple hardware and software issues.

Unit 2: Getting Started: Programming (5 lessons)

Students explore the basics of CodeHop by using commands to move and animate sprites, building familiarity with sequencing, motion, and dialogue. As they progress through a story-driven unit, they apply programming concepts such as events and loops to create interactive and animated stories.

Objectives / Topics Covered	● Use basic commands to program sprites. ● Explore the CodeHop interface.
Lessons	**Drawing Tools** ● Create customized sprites and backdrops using drawing tools. **Scout's Programming Expedition Part 1** ● Use basic commands to make a sprite move and talk in a story-driven activity. **Scout's Programming Expedition Part 2** ● Add sprites and create a sequence to animate the next part of the ongoing story. **Scout's Programming Expedition Part 3** ● Animate a Scout story using loops, events, looks, and motion blocks. **Scout's Programming Expedition Part 4** ● Continue building the animated Scout story using events, looks, and motion blocks.

Unit 3: Digital Literacy (8 lessons)

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

Objectives / Topics Covered	● Behave respectfully online. ● Model how computers store information. ● Analyze data. ● Model how networks function.
Lessons	**Spotting Cyberbullying** ● This lesson is coming soon! ● *Standard Met: CS.3.6.1a*

	Digital Responsibility: Getting Started ● This lesson is coming soon! ● *Standard Met: CS.3.6.1c* **Digital Footprint Basics** ● This lesson is coming soon! ● *Standards Met: CS.3.4.1a, CS.3.6.1b* **Cybersecurity Introduction** ● Identify common cyber threats and explain how to stay safe online. ● *Standards Met: CS.3.2.1a, CS.3.2.1b, CS.3.6.1d* **Data Detectives** ● Evaluate data for reliability and to draw conclusions and make predictions. **File Management and Data** ● Explain how digital data varies in size and where different types of files are stored. **Modeling Network Connections** ● Model how messages move between devices and how data travels across a network. ● *Standard Met: CS.3.2.1c* **AI Products: Ethical and Responsible Choices** ● Analyze the benefits and challenges of various AI products. ● *Standards Met: CS.3.5.1a, CS.3.5.1b, CS.3.5.1c*

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	● Apply broadcast messages to enable communication between multiple sprites. ● 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	**Sequences: Parallel Programming** ● Create a program that uses parallel sequences running at the same time. **Events** ● Create a program using events. **Broadcast Messages: Marco Polo** ● Use broadcast messages to trigger actions between sprites. **Loops** ● Explain that a loop repeats actions in a program. **Debugging: Events and Loops** ● Decompose a program in order to debug it. **Loops: Falling Objects (2 part lesson)** ● Compare advantages and disadvantages of different algorithms using loops. ● *Standard Met: CS.3.1.1a* **Introduction to Conditionals** ● Explain what a conditional is in programming and create a program that uses if/then blocks. **Variables** ● Create and change the value of a variable in a program. **Introduction to Comparison Operators** ● Use comparison operators in “if/else” blocks. **Introduction to Functions** ● Create and use functions in a program. **I-Spy with Lists (2 part lesson)** ● Create an I-Spy style activity using variables and lists.

Unit 5: Interdisciplinary Exploration (9 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	**Geographic Effects on Early Civilizations** ● Explain how physical geography influenced early civilizations using broadcast messages. **Communities Adapt to & Modify Their Environment** ● Use click events to create a scene showing how communities adapt to or change their environments. **Weather and Climate** ● Use climate data and event blocks to model typical weather conditions for a selected month. **Exploring Adaptations** ● Create an interactive program showing how animal adaptations support survival in different habitats. **Animating Poetry** ● Break a poem into parts and create an animated reading that blends literal and figurative meaning. **Animating Sprites with Multiplication** ● Use multiplication, loops, and wait blocks to animate sprites. **Adding with Loops** ● Use loops to repeat commands and model multi-digit addition based on place value. **Classifying Shapes by Category** ● Use events to classify quadrilaterals based on defining properties.

3rd 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**.	