Arkansas Computer Science: 1st Grade Course Syllabus

One Year for Elementary School, 36 Hours

Course Overview and Goals

The **Arkansas Computer Science: 1st Grade Course** introduces students to foundational programming concepts through a block-based programming language. Students explore digital literacy and computer science skills while learning to create interactive projects, animations, and games. This course emphasizes creativity, collaboration, and real-world connections, providing students with a strong foundation in computer science concepts and digital literacy.

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.** Instruction follows an "I do, we do, you do" model and incorporates spiral review to reinforce concepts and build confidence over time. Many digital literacy lessons contain unplugged activities, requiring printed handouts and class activities to support hands-on learning.

The course includes a total of **36 lessons,** each approximately 45 minutes long. 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.

Standards Alignment Note: Lessons that list standards in bold parentheses (below) are required to fully meet state computer science standards.

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.

Assessments: Some units conclude with a brief 3 question quiz designed to assess student understanding of the concepts and skills introduced in the unit. Quizzes can be assigned digitally, completed as a QuizBuzz activity, or printed for paper-based completion.

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/AR_1/overview



A clickable PDF can be found at https://codehs.com/AR-K-4Roadmaps

Course Breakdown

Optional Review

In this optional unit, students follow Scout on a story-driven adventure to review logging in, adding characters, backgrounds, and pages, and building sequences with motion blocks.

Objectives / Topics Covered	● Log in and navigate the Playground ● Add and modify characters, backgrounds, and pages ● Build sequences with motion blocks and events
Lessons	**Welcome to CodeHop!** ● Learn how to log in and explore the Playground to prepare for future lessons. **Scout Adventures 1: Introducing Scout** ● Explore the programming interface and add characters to a 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 in a program. **Scout Adventures 4: Scout Explores the Forest** ● Use motion blocks to move characters around the stage. **Scout Adventures 5: Scout and Bluebird Help** ● Build a sequence of motion blocks to help characters collect objects. **Scout Adventures 6: Scout Celebrates with Friends** ● Create a celebration scene using characters, pages, backgrounds, and sequences with events.

Unit 1: Getting Started (3 weeks)

In this unit, students use computational thinking to break down their evening routines and are introduced to sequencing and loops through unplugged programming activities.

Objectives / Topics Covered	● Identify patterns and break down everyday tasks ● Create and sequence block-coding programs ● Repeat simple sequences with loops
Lessons	**Computational Thinking: Evening Routines (CSK4.1.2.1.1, CSK4.1.2.1.2, CSK4.1.2.1.3, CSK4.1.2.1.4)** ● Identify patterns, break down tasks, and sequence the steps of an evening routine. **Unplugged Introduction to Programming: Part 1 (CSK4.1.2.2.1)** ● Create and sequence block-coding programs without a computer. **Unplugged Introduction to Programming: Part 2 (CSK4.1.2.2.1)** ● Use loops to repeat simple sequences in an unplugged activity. **Getting Started Quiz (3 questions)**

Unit 2: Sequences & Events (4 weeks)

In this unit, students use events and sequences to program characters, find and fix bugs, and practice the design process to solve a real-world problem.

Objectives / Topics Covered	● Use multiple event blocks in a program ● Find and correct bugs in sequences ● Plan, create, and revise a project using the design process
Lessons	**Events (CSK4.1.2.1.2)** ● Explain what an event is and use multiple event blocks in a program.

	Sequences: Digital Responsibilities (CSK4.1.2.3.2) ● Program characters with sequences to show how to be responsible online. **Introduction to Debugging (CSK4.1.2.2.2, CSK4.1.4.2.2)** ● Find and correct bugs in sequences. **Practicing the Design Process (CSK4.1.2.1.3, CSK4.1.2.1.4)** ● Plan, create, and revise a program that models a solution to a simple real-world problem. **Sequences & Events Quiz (3 questions)**

Unit 3: Loops & Message Events (5 weeks)

In this unit, students use repeat and forever loops to simplify their code and send messages so characters can communicate with each other.

Objectives / Topics Covered	● Use repeat and forever loops ● Use message events to make characters interact
Lessons	**Coding Card Game: Loops (CSK4.1.2.2.1)** ● Work together to build instructions with loops that move Scout through a maze. **Introduction to Repeat Loops (CSK4.1.2.1.1)** ● Use repeat loops to run a section of code multiple times. **Forever Loop Dance Party (CSK4.1.2.3.2)** ● Use a “repeat forever” loop to make characters dance. **Introduction to Message Events (CSK4.1.2.3.2)** ● Program a relay race that uses messages to make characters interact. **Message Events: Simon Says (CSK4.1.2.3.2)** ● Use message events so one character can communicate with many characters. **Loops & Message Events Quiz (3 questions)**

Unit 4: Pages & Grid (4 weeks)

In this unit, students build games and projects that move between pages, use the grid to navigate mazes, and end programs with the end block.

Objectives / Topics Covered	● Use “go to page” blocks to move between pages ● Use the grid to move characters to specific locations ● Use the end block to stop a program
Lessons	**Pages: Create a Tapping Game (CSK4.1.2.3.2)** ● Create a game that moves from page to page using “go to page” blocks. **Digital Greeting Card (CSK4.1.4.2.1)** ● Create a digital greeting card using events and loops. **Grid: Solving Mazes (CSK4.1.2.3.2)** ● Design a maze and use the grid to program a character through it. **End Block: Program a Race (CSK4.1.2.3.2)** ● Program a race and use the “end” block to finish the animation. **Pages & Grid Quiz (3 questions)**

Unit 5: Culmination Projects (5 weeks)

In this unit, students plan and build personal and game projects that combine events, loops, and speed blocks.

Objectives / Topics Covered	● Plan a program before building it ● Program obstacles and adjust difficulty in a game

Lessons	**About Me Project (3 part lesson | CSK4.1.2.1.3, CSK4.1.2.1.4)** ● Plan and create a program that describes their characteristics and interests. **River Crossing Game (2 part lesson | CSK4.1.2.3.2)** ● Program obstacles in a game and change the difficulty using speed blocks.

Unit 6: Productivity Software (2 weeks)

In this unit, students use word processing and presentation tools to create documents and slides about their favorite things.

Objectives / Topics Covered	● Use word processing tools to create a document ● Use presentation software to create slides
Lessons	**Introduction to Word Processing (CSK4.1.4.2.1, CSK4.1.4.2.2, CSK4.1.4.2.3)** ● Use word processing tools to write about their favorite season. **Introduction to Presentations (CSK4.1.4.2.1, CSK4.1.4.2.3)** ● Use presentation software to create slides about their favorite things. **Productivity Software Quiz (3 questions)**

Unit 7: Computing Systems (2 weeks)

In this unit, students explore the parts of a computer and learn how networks and the Internet help people communicate.

Objectives / Topics Covered	● Identify input, output, hardware, and software ● Explain what a network is ● Model how messages travel over the Internet
Lessons	**Computer Basics: Exploration (CSK4.1.2.3.1, CSK4.1.4.1.1, CSK4.1.4.1.3, CSK4.1.4.1.4)** ● Identify input, output, hardware, and software and what to do when a computer doesn't work. **Networks and the Internet (CSK4.1.4.3.1, CSK4.1.4.3.2, CSK4.1.4.3.3)** ● Explain what a network is and model how messages travel over the Internet. **Computing Systems Quiz (3 questions)**

Unit 8: Safety & Responsibility (3 weeks)

In this unit, students identify positive online behavior and practice using technology and websites safely and responsibly.

Objectives / Topics Covered	● Identify appropriate and inappropriate online behavior ● Use technology safely and responsibly ● Recognize safe websites
Lessons	**Positive Online Behavior (CSK4.1.1.1.1, CSK4.1.1.1.2, CSK4.1.1.1.3, CSK4.1.1.1.4, CSK4.1.1.2.1)** ● Identify appropriate and inappropriate online behavior in a program. **Exploring Responsible Technology Use (CSK4.1.1.1.1, CSK4.1.1.3.1, CSK4.1.4.1.2)** ● Explain ways to use technology safely and responsibly. **Exploring Website Safety (CSK4.1.1.2.2, CSK4.1.1.2.3)** ● Recognize safe websites and practice protecting personal information online. **Safety & Responsibility Quiz (3 questions)**

Unit 9: Impacts of Computing (2 weeks)

In this unit, students explore how computing devices shape daily life and how computer science can help solve community problems.

Objectives / Topics Covered	● Identify computing devices in daily life ● Explain how technology has changed how people live, work, and play ● Connect computer science to community problems
Lessons	**Impacts of Computing: Our Community (CSK4.1.1.3.1)** ● Identify computing devices in daily life and explain how technology has changed how people live, work, and play. **Careers in CS: Litter Free Communities (CSK4.1.2.3.2)** ● Create a program to collect and sort litter and explain how computer science helps communities. **Impacts of Computing Quiz (3 questions)**

Unit 10: Data & Analysis (4 weeks)

In this unit, students conduct guided research, collect and organize data, and learn how computers store data as files.

Objectives / Topics Covered	● Find information using research sources ● Collect, organize, and display data ● Recognize that computers store data as files
Lessons	**Guided Research (2 part lesson | CSK4.1.1.3.2)** ● Find information using research sources and create a program to share their findings. **Basic Data and Programming (CSK4.1.3.1.1, CSK4.1.3.1.2, CSK4.1.3.1.3, CSK4.1.3.2.3, CSK4.1.3.3.1, CSK4.1.3.3.2)** ● Collect and organize data, then use a program to display it and answer questions. **Data Storage and Files Practice (CSK4.1.4.2.3)** ● Model how data is collected and stored as files on a computer. **Data & Analysis Quiz (3 questions)**

Unit 11: Artificial Intelligence (2 weeks)

In this unit, students identify tools that use AI and explore how patterns and data help computers make decisions.

Objectives / Topics Covered	● Identify tools that use AI ● Compare tasks suited for humans and AI ● Use patterns to classify objects
Lessons	**What Can AI Do?** ● Identify tools that use AI and compare tasks better suited for humans or AI. **Sorting with Charts (CSK4.1.3.2.1, CSK4.1.3.2.2)** ● Use a decision tree to classify objects and create a chart showing how patterns help computers decide. **Artificial Intelligence Quiz (3 questions)**

1st 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.
These resources and more are found on the **CodeHop Resources Page**.	