Alabama Digital Literacy and Computer Science: 2nd Grade Course Syllabus

One Year for Elementary School, 36 Hours

Course Overview and Goals

The **Alabama Digital Literacy and Computer Science: 2nd 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 Met*" (below) are required to fully meet state 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/AL_2/overview

A clickable PDF can be found at https://codehs.com/AL-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. ● Reinforce understanding of basic programming concepts such as events, loops, and message blocks.
Lessons	**Welcome to CodeHop!** ● Learn how to log in and use the Playground; a quick intro lesson to get started with the platform. **Introduction to CodeHopJr** ● Explore the CodeHopJr interface and create a scene by adding and placing characters. **Events** ● Learn what an event is in programming and use event blocks to trigger actions in a project. **Introduction to Repeat Loops** ● Use repeat loops to run a section of code multiple times and simplify repeated actions. **Forever Loop Dance Party** ● Create a fun animation where characters repeat movements continuously using the “repeat forever” loop. **Introduction to Message Events** ● Build a relay-style animation where characters interact using message blocks to control the flow of the program.

Unit 1: Getting Started (3 weeks)

In this unit, students will explore the basics of computing by learning the functions of computers and their components, practice identifying positive and negative online behaviors, and apply foundational computational thinking skills like pattern recognition, sequencing, and task decomposition to real-life routines.

Objectives / Topics Covered	● Understand the basic components of a computer. ● Learn responsible ways to use technology. ● Apply computational thinking in everyday routines.
Lessons	**Computer Basics: Connections** ● Learn what a computer is, how we use it, and what to do when it doesn’t work. ● *Standards Met: DLCS25.2.10, DLCS25.2.11, DLCS25.2.17* **Practicing Responsible Technology Use** ● Explore ways to use technology safely, respectfully, and responsibly in daily life. ● *Standard Met: DLCS25.2.9* **Computational Thinking: School Day Routines** ● Use computational thinking to find patterns, break down routines, and organize steps to better understand everyday school activities. ● *Standard Met: DLCS25.2.1*

Unit 2: Sequences and Events (6 weeks)

In this module, students build foundational programming skills by creating and adjusting algorithms, using the grid to control movement, and designing interactive programs with events.

Objectives	● Develop simple algorithms to control character movement and behavior.

/ Topics Covered	• Use events to trigger actions. • Apply grid-based positioning to guide character actions.
Lessons	**Debugging: Events and Sequences** • Find and fix errors in a program. • *Standard Met: DLCS25.2.2* **Introduction to the Grid** • Use the grid feature to move characters accurately to specific locations on the page. **Impacts of Computing: Changing World** • Explain how technology has changed the way people live, work, and play. • *Standard Met: DLCS25.2.12, DLCS25.2.14* **Tap-a-Mole Game (2 part lesson)** • Design an interactive game using event blocks to trigger actions when characters are tapped. **Careers in CS: Litter Free Communities** • Explain how computer science can help solve community problems by creating a program to collect and sort litter. • *Standard Met: DLCS25.2.4*

Unit 3: Message Events (3 weeks)

Students will deepen their understanding of program control by using message events to trigger actions, model cycles, and coordinate activity across multiple pages.

Objectives / Topics Covered	• Use message events to control the flow of a program. • Navigate between pages in a program.
Lessons	**Message Events: Scout Plays in the Forest** • Use message events to control when and how different parts of a program run. **Programming a Cycle** • Model a real-world cycle by using message blocks to create repeating sequences in a program. • *Standard Met: DLCS25.2.4* **Pages: Scout's Travels** • Use message events to move between pages in a multi-scene program.

Unit 4: Loops (4 weeks)

Students will apply loops to identify patterns, build timers, and simplify repeated actions in code.

Objectives / Topics Covered	• Use loops to simplify repeated actions in a program.
Lessons	**Loops: Follow the Path** • Identify patterns and use loops to create a program that repeats actions along a path. • *Standard Met: DLCS25.2.1* **Making a Timer** • Use loops, wait blocks, and turn blocks to create and compare timers with different speeds. **Algorithms: Connecting a Path** • Adjust simple algorithms to move characters based on size, shape, and starting position. • *Standard Met: DLCS25.2.4* **Exploring the Design Process** • Use the design process to improve a program that models a solution to a real-world problem. • *Standards Met: DLCS25.2.1, DLCS25.2.18*

Unit 5: Culmination Projects (7 weeks)

In this unit, students apply their full range of computer science skills to create interactive games and animations using sequences, events, loops, messages, and pages.

Objectives / Topics Covered	● Design interactive games that incorporate events, loops, messages, and sequences. ● Demonstrate mastery of computer science concepts through original projects. ● Collaborate with other classrooms digitally.
Lessons	**Code Block Review (2 part lesson)** ● Use a variety of coding blocks in a program and explain what each one does. **Maze Game Project (3 part lesson)** ● Create and explore multiple ways to create an interactive game. **Collaborating Digitally (2 part lesson)** ● Collaborate with others digitally to create a program about different classrooms. ● *Standard Met: DLCS25.2.9*

Unit 6: Productivity Software (3 weeks)

In this unit, students will practice creating digital projects using word processing, presentation, and spreadsheet software.

Objectives / Topics Covered	● Create and edit a document. ● Create a presentation. ● Add data to a spreadsheet to visualize data.
Lessons	**Exploring Word Processing** ● Create a document explaining a favorite season. ● *Standard Met: DLCS25.2.10* **Exploring Presentations** ● Create a presentation on favorite things. ● *Standard Met: DLCS25.2.10* **Exploring Spreadsheets** ● Present data on class pets and favorite colors. ● *Standard Met: DLCS25.2.10*

Unit 7: Digital Literacy (10 weeks)

In this unit, students explore and practice digital literacy concepts to gain an understanding of current and emerging technologies.

Objectives / Topics Covered	● Understand the importance of secure login information. ● Collect, analyze, and visually present data. ● Conduct research using reliable sources and share findings visually. ● Explore how computer networks function.
Lessons	**Keyboarding Basics (Ongoing typing program)** ● Use a keyboard to type simple words and sentences. ● *Standard Met: DLCS25.2.20* **Responsible Digital Citizens** ● Explain what it means to be responsible digital citizens. ● *Standards Met: DLCS25.2.9, DLCS25.2.16* **Password Protectors** ● Learn why usernames and passwords are important and practice ways to keep login information safe. ● *Standard Met: DLCS25.2.8*

	Advanced Data and Programming (3 part lesson) ● Ask a question, collect data through a survey, and create a program to present the results visually. ● *Standards Met: DLCS25.2.3, DLCS25.2.5, DLCS25.2.18* **Choice Research (2 part lesson)** ● Research a topic using trusted sources and create a visual program to share findings. ● *Standards Met: DLCS25.2.9, DLCS25.2.15, DLCS25.2.18* **Exploring Computer Networks** ● Model sending and receiving information using message blocks. ● *Standard Met: DLCS25.2.7* **What Can AI Do?** ● Compare tasks that are better suited for humans vs. AI. ● *Standards Met: DLCS25.2.13, DLCS25.2.19* **Machine Learning: AutoDraw** ● Describe how AutoDraw uses AI to recognize and suggest drawings. ● *Standards Met: DLCS25.2.3, DLCS25.2.6*

Preparing for Next Year

In this optional unit, students expand their programming knowledge by using conditionals in unplugged activities and transitioning from CodeHopJr to CodeHop.

Objectives / Topics Covered	● Use conditionals to guide character action. ● Transition from CodeHopJr to CodeHop. ● Apply familiar programming concepts, such as events and loops, in a new coding environment.
Lessons	**Coding Card Game: Conditionals** ● Work together to create a sequence of instructions using conditionals to guide a character through a maze. **Coding Card Game: Conditionals 2** ● Continue practicing conditionals by solving new maze challenges with team-based instruction sequences. **From CodeHopJr to CodeHop Blocks** ● Explore the basic interface of the CodeHop editor and create a simple program using familiar concepts.

2nd 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**.	