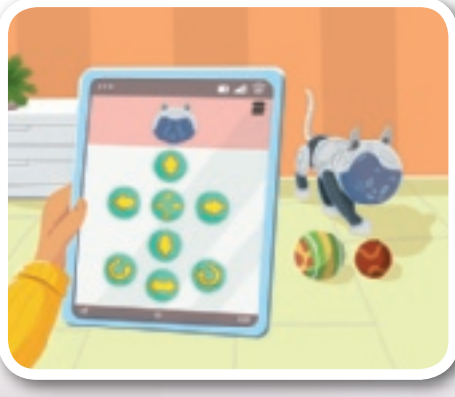
Core Knowledge® SCIENCE

# Codes and Computers

## Teacher Guide

Using technology

Commands



Programming

# Codes and Computers

## Teacher Guide

ISBN: 978-1-68380-992-0

# Codes and Computers

## Table of Contents

# Codes and Computers

## Teacher Guide

**Core Knowledge Computer Science™ 3**

GRADE 3

# Introduction

## About this Program

### Core Knowledge Computer Science Conceptual Approach Framework

| Grade | Title and Driving Question | Emphasis is on . . . | Overarching Unit Objective |
|---|---|---|---|
| K (ages 5–6) | **Computers All Around Us**<br>What are computers and where can we find them? | recognizing | Identify computer devices and parts and operate them age-appropriately. |
| 1 (ages 6–7) | **Helpful Computers**<br>What can computer programs do? | using | Experience a variety of age-appropriate programs designed for different purposes. |
| 2 (ages 7–8) | **Using Computers**<br>How can we use computers and the internet? | using | Use a variety of age-appropriate programs and websites with increasing skill and independence under supervision. |
| 3 (ages 8–9) | **Codes and Computers**<br>How do programmers build computer programs? | deciphering and evaluating | Model use of symbols, codes, and steps in simple processes and problem solving. |
| 4 (ages 9–10) | **Problem-Solving and Computers**<br>What kinds of problems can we solve with computers? | deciphering and evaluating | Model solution design in increasing complexity and involving conditionals and loops. |
| 5 (ages 10–11) | **Designing Computer Programs**<br>What kinds of computer programs can I develop? | planning and designing | Write, evaluate, and debug code to execute grade-appropriate tasks. |

### Skills and Performance Summary

**Grade K**

The goal of the instruction and cumulative experiences in this unit is for young students to build a concept of what makes a thing a computer. Students think about how tools and devices can be helpful for completing tasks. Though students will not specifically study simple and compound machines in science until a later grade level,

they are able from everyday exposure to differentiate devices that need a power source and those that do not. Within the class of devices that need a power source, students can form a further classification of electronic devices—ones that do something with information that is put into the device. Upon recognizing input-processing-output characteristics of devices, students can also begin to classify and use devices for different purposes, including playing games, communicating, and doing schoolwork.

**Grade 1**

Building on the recognition established in the previous grade, the goal of the instruction and cumulative experiences in this unit is for young students to begin using computer devices for various purposes, including game play, communication, and schoolwork. Through practice, students understand that the operation of devices has a series of steps in common, in which users input information, the device performs a process, and then a useful output occurs. Inputs include the use of a mouse, keyboard, touchscreen, game controller, camera, and microphone. Output includes screen visuals (text, still images, video), hard-copy printout, and audio. Accessing desired outputs from computers depends first on successful operation of input devices.

**Grade 2**

Building on the use of devices established in the previous grade, the goal of the instruction and cumulative experiences in this unit is for young students to continue using with increasing independence computer devices for various purposes, including game play, communication, and schoolwork. With increasing practice in successfully inputting information, students can consider ways to achieve creative outputs that use and combine text, graphics, visual data displays, photos, animations, video, and audio.

**Grade 3**

In this grade level, the emphasis begins to shift from use—skillfully inputting information to prompt outputs—to the unseen processes inside computing technology. The goal of the instruction and cumulative experiences in this unit is for students to identify the output of every computer device as a solution to a problem and deconstruct the solution into the necessary steps required. Students discern that computers do not think on their own to solve problems but only execute steps-by-step instructions that are built into them. And they begin to dissect messages into the granular parts that make them up and transmit them.

**Grade 4**

Building on the concepts of symbols, codes, signals, and step-by-step instructions established in the previous grade, the goal of the instruction and cumulative experiences in this unit is for students to continue articulating steps of increasingly complexity to solve problems. Beyond dissecting solutions into granular steps, students think forward in scenarios with conditional branching options (if-then statements) and loops (if-then, repeat prior instruction). Students relate what computing devices do to their programming, and they build understanding that complex computer programming is the outcome of building, layering, and chaining together links of fundamentally simple code.

**Grade 5**

In this grade level, the emphasis begins to shift from examination—figuring out how instructions make computing devices do what they do—to creative application of that knowledge. They step forward from the role of computer technology user into a beginning maker role. Students begin to use functional computer language to craft code for executable tasks.

## Note to Teachers and Curriculum Planners

This unit introduces Grade 3 students to fundamental concepts related to computer programming. The unit includes a story-based Student Reader and this Teacher Guide. The lesson sessions alternate between those focused on the Student Reader and others that are interactive in nature and do not involve a Student Reader chapter.

This unit can be integrated into your existing science or math curriculum. The lessons can be inserted intermittently, or the entire unit can be taught on consecutive days. The unit can also be offered as an enrichment course.

## Standards: What are the relevant CSTA Concepts for this unit?*

This unit, *Codes and Computers*, has been informed by the K–12 Computer Science Standards put forth by the Computer Science Teachers Association (CSTA). The CSTA K–12 Computer Science Standards delineate a core set of learning objectives designed to provide the foundation for computer science curricula and implementation at the K–12 level. The CSTA Standards introduce fundamental concepts of computer science beginning at the elementary school level.

**Online Resources**

For detailed information about the CSTA references, follow the links in the Online Resources Guide for this unit:

**www.coreknowledge.org/cksci-online-resources**

### Grades 3–5 CSTA Standards

| Concepts | Subconcepts | Level 1B (ages 8–11)<br>By the end of Grade 5, Students will be able to . . . |
|---|---|---|
| **Computing Systems** | **Devices** | 1B-CS-01 Describe how internal and external parts of computing devices function to form a system. (P7.2) |
| | **Hardware and Software** | 1B-CS-02 Model how computer hardware and software work together as a system to accomplish tasks. (P4.4) |
| | **Troubleshooting** | 1B-CS-03 Determine potential solutions to solve simple hardware and software problems using common troubleshooting strategies. (P6.2) |

* The K–12 Computer Science Framework, led by the Association for Computing Machinery, Code.org, Computer Science Teachers Association, Cyber Innovation Center, and National Math and Science Initiative in partnership with states and districts, informed the development of this work. Authors of the CSTA K–12 Computer Science Standards were not involved in the production of this product, and their endorsement is not implied.

**Source:**

Computer Science Teachers Association (2017). CSTA K–12 Computer Science Standards, Revised 2017. Retrieved from http://www.csteachers.org/standards.

K–12 Computer Science Framework. (2016). *Framework view by grade band*. Retrieved from http://www.k12cs.org

| Concepts | Subconcepts | Level 1B (ages 8–11)<br>By the end of Grade 5, Students will be able to . . . |
|---|---|---|
| **Networks and the Internet** | **Network, Communication, and Organization** | 1B-NI-04 Model how information is broken down into smaller pieces, transmitted as packets through multiple devices over networks and the Internet, and reassembled at the destination. (P4.4) |
| | **Cybersecurity** | 1B-NI-05 Discuss real-world cybersecurity problems and how personal information can be protected. (P3.1) |
| **Data and Analysis** | **Storage** | (continuation) 1A-DA-05 Store, copy, search, retrieve, modify, and delete information using a computing device and define the information stored as data. (P4.2) |
| | **Collection, Visualization, and Transformation** | 1B-DA-06 Organize and present collected data visually to highlight relationships and support a claim. (P7.1) |
| | **Inference and Models** | 1B-DA-07 Use data to highlight or propose cause-and-effect relationships, predict outcomes, or communicate an idea. (P7.1) |
| **Algorithms and Programming** | **Algorithms** | 1B-AP-08 Compare and refine multiple algorithms for the same task and determine which is the most appropriate. (P6.3, P3.3) |
| | **Variables** | 1B-AP-09 Create programs that use variables to store and modify data. (P5.2) |
| | **Control** | 1B-AP-10 Create programs that include sequences, events, loops, and conditionals. (P5.2) |
| | **Modularity** | 1B-AP-11 Decompose (break down) problems into smaller, manageable subproblems to facilitate the program development process. (P3.2) |
| | | 1B-AP-12 Modify, remix, or incorporate portions of an existing program into one's own work, to develop something new or add more advanced features. (P5.3) |
| | **Program Development** | 1B-AP-13 Use an iterative process to plan the development of a program by including others' perspectives and considering user preferences. (P1.1, P5.1) |
| | | 1B-AP-14 Observe intellectual property rights and give appropriate attribution when creating or remixing programs. (P7.3) |
| | | 1B-AP-15 Test and debug (identify and fix errors) a program or algorithm to ensure it runs as intended. (P6.1, P6.2) |
| | | 1B-AP-16 Take on varying roles, with teacher guidance, when collaborating with peers during the design, implementation, and review stages of program development. (P2.2) |
| | | 1B-AP-17 Describe choices made during program development using code comments, presentations, and demonstrations. (P7.2) |

| Concepts | Subconcepts | Level 1B (ages 8–11)<br>By the end of Grade 5, Students will be able to . . . |
|---|---|---|
| **Impacts of Computing** | **Culture** | 1B-IC-18 Discuss computing technologies that have changed the world, and express how those technologies influence, and are influenced by, cultural practices. (P7.1) |
| | | 1B-IC-19 Brainstorm ways to improve the accessibility and usability of technology products for the diverse needs and wants of users. (P1.2) |
| | **Social Interactions** | 1B-IC-20 Seek diverse perspectives for the purpose of improving computational artifacts. (P1.1) |
| | **Safety, Ethics, and Law** | 1B-IC-21 Use public domain or creative commons media, and refrain from copying or using material created by others without permission. (P7.3) |

**Integrated Practices**

1. Fostering an Inclusive Computing Culture
    - 1.1 Include the unique perspectives of others . . .
    - 1.2 Address the needs of diverse end users . . .
    - 1.3 Employ self and peer-review advocacy . . .
2. Collaborating Around Computing
    - 2.1 Cultivate working relationships . . .
    - 2.2 Create team norms, expectations, and equitable workloads . . .
    - 2.3 Solicit and incorporate feedback . . .
    - 2.4 Evaluate and select technological tools . . .
3. Recognizing and Defining Computational Problems
    - 3.1 Identify complex, interdisciplinary, real-world problems . . .
    - 3.2 Decompose complex real-world problems . . .
    - 3.3 Evaluate whether it is appropriate and feasible . . .
4. Developing and Using Abstractions
    - 4.1 Extract common features . . .
    - 4.2 Evaluate existing technological functionalities . . .
    - 4.3 Create modules and develop points of interaction . . .
    - 4.4 Model phenomena and processes and simulate systems . . .

5. Creating Computational Artifacts

    5.1 Plan the development of a computational artifact . . .

    5.2 Create a computational artifact . . .

    5.3 Modify an existing artifact . . .

6. Testing and Refining Computational Artifacts

    6.1 Systematically test . . .

    6.2 Identify and fix errors . . .

    6.3 Evaluate and refine . . .

7. Communicating About Computing

    7.1 Select, organize, and interpret . . . data sets . . .

    7.2 Describe, justify, and document computational processes . . .

    7.3 Articulate ideas responsibly . . .

(See *K12 Computer Science Framework* for full text of the Practices integrated into the standards.)

## What Teachers Need to Know

Supportive information on the content standards and the concepts they address is provided throughout the lessons at points of relevance:

**Know the Standards:** These sections, found later in this Teacher Guide, explain what to teach and why, with reference to CSTA and Core Knowledge expectations.

**Know the Science:** These sections provide supporting, adult-level, background information or explanations related to specific computer science concepts or examples.

## A Special Note to Grade 3 Teachers Before Starting This Unit

### Why Study Computer Science in the Elementary Grades?

For many elementary teachers, the suggestion that computer science should be a part of the Grade 3 curriculum may seem questionable. For many teachers, precious instructional time is devoted to teaching core content area fundamentals to students.

The study of computer science in Grade 3, however, is consistent with the Core Knowledge approach to learning. Knowledge builds upon knowledge, and computer science builds upon the foundational understanding that many devices share basic underlying technology. And that technology operates through shared processes that are simple at their core but are combined in increasingly complex ways that can produce unlimited outcomes. Grade 3 students can build on what they learned in Kindergarten through Grade 2, to recognize that devices are

operated by instructional coding. They can also consider step-by-step instructions, if-then scenarios, and simple codes that carry meaning. These building blocks promote readiness for students to put concepts together and make their own instructional codes in later elementary grades.

## Using the Student Reader

Student Reader

The *Codes and Computers* Student Reader includes nine chapters and a student Glossary providing definitions to Core Vocabulary words. Engaging text and images encourage students to draw upon their own experiences and the world around them to understand unit concepts. In addition to Core Vocabulary, the Student Readers include a feature called Word to Know, which provides background information to help students understand key terms, and may sometimes include additional informational boxes, such as Think About.

**Read, then explore:** In the Core Knowledge Computer Science program, lessons are sequenced for reading before active engagement. First, students are introduced to concepts through relatable scenarios in the Student Readers. Then students explore the concepts through hands-on investigations or teacher demonstrations, accompanied by active questioning and analysis. The icon shown at left will signal Core Lesson segments that focus on Student Reader chapters.

**Independent reading or group read aloud:** While the text in the Student Readers is written for independent reading, we encourage group read aloud and engagement with the text. The Teacher Guide provides Guided Reading Supports to prompt discussion, clarify misconceptions, and promote understanding in relation to the Driving Question and chapter topics.

Online Resources

Use this link to download the Online Resources for this unit, where the specific link to this appendix can be found:

**www.coreknowledge.org/cksci-online-resources**

## Using the Teacher Guide

### Instructional Design

The *Codes and Computers* unit is the Grade 3 installment in the Core Knowledge Computer Science elementary series. To meet CSTA Standards we encourage teachers to complete the corresponding grade-level unit each school year. Each lesson part requires thirty to forty-five minutes of instruction time. The time it takes to complete a full lesson depends on class size and individual circumstances.

Within the Teacher Guide, each Core Lesson is generally composed of multiple two or three parts. Each segment concludes with a Check for Understanding, providing the teacher with a quick opportunity for formative assessment. At the end of this unit introduction, you will find a blank pacing guide, which you may use to plan how you might pace the lessons. We strongly recommend that you preview the unit in full before beginning and create your pacing guide before teaching the first lesson.

## Features

The unit is composed with several integrated features that support learning for all students and development for teachers.

| | |
|---|---|
| **Unplugged**<br>  <br>**Device-Based**<br>  <br>**Online Resources**<br>  | This computer science unit can be completedin classrooms that have computers and devices for students and those that do not. Look for the icons at left to indicate the types of student interactions recommended within each lesson. Pay attention to those icons most relevant to your classroom situation.<br><br>There are many opportunities for students to complete foundational exercises relevant to the concepts of computer science without needing access to computers. These activities are designated by the Unplugged icon.<br><br>Several activities and extensions do require that students work on computers, and those experiences are designated by the Device-Based icon.<br><br>A few activities and demonstrations recommended in this unit require the use of internet-connected devices. These experiences are indicated by the Online icon and will also include references to the Online Resources Guide for links to the recommended internet resources. |
| **Differentiation** | Adjustments to instruction appear in the text, indicated by **SUPPORT**, **EXTEND**, and **CHALLENGE** notations.<br><br>**SUPPORT**—Reading, writing, listening, and/or speaking alternatives appear for students who are English language learners, have special needs, or read below the grade level. Extra support is suggested for students who struggle to meet targeted expectations.<br><br>**EXTEND**—Extensions are suggested for students with high interest or who have already met the performance expectations.<br><br>**CHALLENGE**—Additional, relevant, and interesting exercises are suggested for students to explore that exercise math, reading, or science skills/comprehension that pushes beyond the grade level. |
| **Teacher Development** | Information in the instructional text, Know the Science boxes, and Know the Standards boxes is provided to support ongoing teacher development with regard to both content and the teaching process. |
| **Monitor Progress** | Opportunities for formative assessment appear throughout the instructional support. These instances are most consistently noted in a Check for Understanding that concludes each lesson segment. |

The K–12 Computer Science Framework, led by the Association for Computing Machinery, Code.org, Computer Science Teachers Association, Cyber Innovation Center, and National Math and Science Initiative in partnership with states and districts, informed the development of this work.

## The Core Lessons

- Lesson time: Each Core Lesson part constitutes one classroom session of thirty to forty-five minutes. Some activities and performance tasks will require setting aside a longer block of time.
- Lesson order: The lesson parts are coherently sequenced to build from one to the next, linking student engagement across lessons and helping students build new learning on prior knowledge.

| **Driving Question:** How do programmers build computer programs? | | |
|---|---|---|
| **Overarching Unit Objective:** Model use of symbols, codes, signals, and steps in simple processes and problem-solving. | | |
| **Lesson and Part** | **Title** | **Interaction Type and Learning Objective** |
| **Lesson 1, Part 1** | What Is Computer Hardware? | Student Reader Read Aloud: Chapter 1, "What Is Computer Hardware?"<br>• Identify the basic hardware components of computer devices.<br>• Define *software* and describe apps suited for different purposes. |
| **Lesson 1, Part 2** | What Are the Components of a Computing System? | Classroom Interactive<br>• Identify icons used to represent different applications.<br>• Locate and open specific software to complete an activity.<br>• Restart a computer or device when a computing issue arises. |
| **Lesson 2, Part 1** | What Are Computer Networks? | Student Reader Read Aloud: Chapter 2, "What Are Computer Networks?"<br>• Define *network*.<br>• Describe ways that people can connect to information and to other people through a network.<br>• Articulate the importance of behaving responsibly and respectfully of others online. |
| **Lesson 2, Part 2** | What Is the Internet? | Classroom Interactive<br>• Describe how the internet allows people in different places to communicate.<br>• Identify physical components required for internet access. |
| **Lesson 3, Part 1** | What Should I Share Online? | Student Reader Read Aloud: Chapter 3, "What Should I Share Online?"<br>• Identify what information is safe and appropriate to share online.<br>• Recognize why safety is important when using computing devices and the internet. |

| | | |
|---|---|---|
| **Lesson 3, Part 2** | What Is a Digital Footprint? | Classroom Interactive<br>• Explain the concept of a digital footprint.<br>• Understand that login information should be kept private. |
| **Lesson 4, Part 1** | What Is Digital Data? | Student Reader Read Aloud: Chapter 4, "What Is Digital Data?"<br>• Define the terms *binary code*, *binary digit*, and *bit*.<br>• Recognize that computers use binary numbers to represent information. |
| **Lesson 4, Part 2** | How Can 1s and 0s Make Letters? | Classroom Interactive<br>• Encode letters into binary data and decode binary data back to letters. |
| **Lesson 5, Part 1** | How Can Data Be Organized? | Student Reader Read Aloud: Chapter 5, "How Can Data Be Organized?"<br>• Classify items and information by characteristics. |
| **Lesson 5, Part 2** | How Can Data Be Used to Make Predictions? | Classroom Interactive<br>• Collect and categorize data in up to three categories.<br>• Display data in a visual format.<br>• Identify patterns in data to make a prediction. |
| **Lesson 6, Part 1** | Symbols to Communicate | Student Reader Read Aloud: Chapter 6, "Symbols to Communicate"<br>• Identify ways in which information can be represented by alphanumeric symbols and pictures. |
| **Lesson 6, Part 2** | What Does the Symbol Mean? | Classroom Interactive<br>• Define *symbol*.<br>• Model the way programs store and manipulate data. |
| **Lesson 7, Part 1** | Computer Devices Follow Steps | Student Reader Read Aloud: Chapter 7, "Computer Devices Follow Steps"<br>• Define *algorithm* as step-by-step instructions used to complete a task. |
| **Lesson 7, Part 2** | How Do You Make an Algorithm? | Classroom Interactive<br>• Identify a real-world algorithm.<br>• Construct an algorithm to complete a task. |
| **Lesson 8, Part 1** | Steps Can Repeat | Student Reader Read Aloud: Chapter 8, "Steps Can Repeat"<br>• Describe sequences.<br>• Recognize that sequences can be repeated in loops. |
| **Lesson 8, Part 2** | Sequences and Loops | Classroom Interactive<br>• Create a program with a sequence of commands and simple loops of repeating steps. |

| | | |
|---|---|---|
| **Lesson 9, Part 1** | Series of Commands | Student Reader Read Aloud: Chapter 9, "Series of Commands"<br>• Identify steps computer programmers use in the design process. |
| **Lesson 9, Part 2** | Fun Programming | Classroom Interactive<br>• Use an online application to compose a sequence of commands to program actions of animated characters. |
| **Lesson 9, Part 3** | Solving Design Programs | Classroom Interactive<br>• Identify a problem or task and break it into incremental steps.<br>• Apply the steps of the design process to solve a problem. |

## Activity Pages

Activity Page

AP 1.2

Blackline reproducible masters for Activity Pages are included in Teacher Resources at the back of the Teacher Guide. The icon shown to the left appears throughout the Teacher Guide wherever Activity Pages (AP) are referenced.

Make sufficient copies for your students in advance of each lesson.

Lesson 1, Part 2—Computer Applications (AP 1.2)

## Online Resources for Science

Online Resources

For each CKSci Computer Science, the Teacher Guide includes references to online resources (including external websites and downloadable documents) to enhance classroom instruction. Look for the icon on the left.

Use this link to download the CKSci Online Resources for this unit:

**www.coreknowledge.org/cksci-online-resources**

## Teaching Strategies

**Start with the familiar.**

Lead with an experience. Begin each lesson with a demonstration, activity, or question about a phenomenon to engage students and focus their attention on the topic. Start with the familiar. Every science topic introduced to students relates in some way to their known world and everyday experiences. The purpose of every lesson is to build a bridge between what is familiar to students and broader knowledge about the way the world works.

**Ask driving questions.**

Each multipart lesson contributes to the unit's Driving Question. And at the beginning of each Teacher Guide lesson segment, you will find a reminder of the Overarching Unit Objective. Each lesson activity provides an incremental concept to help students move toward the objective and ultimately be able to offer a confident answer to the Driving Question. Pose discussion questions in each class session with the Driving Question and Overarching Unit Objective in mind.

| | |
|---|---|
| **Encourage scientific thinking.** | Approach the lessons with students not as learning about computer science, but as learning about the world with a scientific mind.<br><br>Throughout the lessons, encourage students to ask questions about what they observe, do, and read. Record relevant questions in a prominent place in the classroom. Guide students back to these questions as opportunities to answer them emerge from readings, demonstrations, and activities. |
| **Use continuous Core Vocabulary instruction.** | During instruction, emphasize Core Vocabulary terms and their meanings in context rather than relying on isolated drill for memorization of definitions. Through scaffolded questioning, encourage students to come up with definitions in their own words and to use the words in their own sentences.<br><br>Core Vocabulary words for each lesson, as well as Language of Instruction, other key terms teachers are encouraged to use in discussing topics with students, are provided at the start of each lesson. You can find Core Vocabulary and selected Language of Instruction definitions in the Glossary at the back of this Teacher Guide. |
| **Emphasize observation and experience.** | Lessons employ various ways for students to learn, including watching, listening, reading, doing, discussing, and writing. |
| **Make frequent connections.** | Use a combination of demonstrations and reading materials, rich with examples, to help students recognize how the concepts they are learning apply in their everyday lives. Prompt students to relate lesson content to their own experiences, to relate the new and unfamiliar to the familiar, and to connect ideas and examples across disciplines. |
| **Monitor student progress.** | Use verbal questioning, student work, and the Check for Understanding prompts at the end of each lesson part to monitor progress during each lesson and to measure understanding at the conclusion of the unit. Many lessons provide tips to help you support students who need further explanations or clarifications. |

## Effective and Safe Classroom and Online Activities

Conducting safe classroom demonstrations and activities and safe online interactions is essential to successful elementary education. The following resources, included at the back of the Teacher Guide, provide Core Knowledge's recommendations for developing effective science and computer science classroom activities:

- Internet Safety
- Student Online Safety Contract
- Strategies for Acquiring Materials

Online Resources

These resources may also be accessed within the CKSci Online Resources Guide for this unit:

**www.coreknowledge.org/cksci-online-resources**

## Materials and Equipment

The unit requires a variety of materials to support various ways of learning (including doing, discussing, listening, watching, reading, and writing). Prepare in advance by collecting the materials and equipment needed for all of the demonstrations and hands-on investigations.

- A large whiteboard, blank sheets of printer paper, student notebooks, pencils, and scissors are routinely required but not listed below.
- A classroom computer, tablet, or other computing device, as needed for discussion, is referred to in the materials for lesson segments in which it is used but is not repeated in the materials listed here. Internet access and the means to project images/videos for whole-class viewing are also required in many lesson segments but not repeated below.

### Lesson 1, Part 2

- large whiteboard
- sticky notes

### Lesson 2, Part 2

- medium paper cups (2 per student)
- string (6 feet per student)
- sush pins to poke holes

### Lesson 3, Part 2

- Large whiteboard
- Print out of the Digital Trails – Animal Tracks Squares

### Lesson 4, Part 2

- Large whiteboard
- Blank paper or student notebooks

### Lesson 5, Part 2

- Large whiteboard
- A minimum of 5 small bags of small, colored candy
- Blank paper

### Lesson 6, Part 2

- blank paper and pencil

### Lesson 7, Part 2

- Printed copies of "Lesson 5- Hello Ruby Map Algorithms"

### Lesson 9, Part 2

- tablets with downloads of the recommended programming app (ScratchJr) for each student (See the online Resources Guide for links to these items. **www.coreknowledge.org/cksci-online-resources**)
- means to project the teacher's screen for demonstration and student guidance
- Images available from https://www.scratchjr.org/pdfs/blocks.pdf
- Interface Guide from https://www.scratchjr.org/pdfs/scratchjr-interface-guide.pdf (optional)

# *Codes and Computers* Pacing

_____**'s Class**

**Note to Teacher:** *Codes and Computers* can be taught on consecutive days as a supplemental unit to your science or math curriculum. It can also be offered as an enrichment program. If the program is implemented in consecutive thirty- to forty-five-minute class sessions, it will take seventeen days to complete.

**Week 1**

| Day 1 | Day 2 | Day 3 | Day 4 | Day 5 |
|---|---|---|---|---|
| | | | | |

**Week 2**

| Day 6 | Day 7 | Day 8 | Day 9 | Day 10 |
|---|---|---|---|---|
| | | | | |

**Week 3**

| Day 11 | Day 12 | Day 13 | Day 14 | Day 15 |
|---|---|---|---|---|
| | | | | |

**Week 4**

| Day 16 | Day 17 | Day 18 | Day 19 | Day 20 |
|---|---|---|---|---|
| | | | | |

LESSON 1, PART 1

# What Is Computer Hardware?

**Driving Question:** How do programmers build computer programs?

**Overarching Unit Objective:** Model use of symbols, codes, signals, and steps in simple processes and problem-solving.

## At a Glance

### Learning Objective

✓ Identify the basic hardware components of computer devices.

### Instructional Activities

- student reading
- whole-class discussion

### CSTA Standards

**Concept:** Computing Systems

**Subconcepts and Standards:**

**Devices: 1B-CS-01** Describe how internal and external parts of computing devices function to form a system. (P7.2)

**Hardware and Software: 1B-CS-02** Model how computer hardware and software work together as a system to accomplish tasks. (P4.4)

**Troubleshooting: 1B-CS-03** Determine potential solutions to solve simple hardware and software problems usingcommon troubleshooting strategies. (P6.2)

For detailed information about the CSTA Standards, follow the links in the Online Resources Guide for this unit:

**www.coreknowledge.org/cksci-online-resources**

## Core Vocabulary and Language of Instruction

A Glossary at the end of this Teacher Guide lists definitions for Core Vocabulary and selected Language of Instruction.

**Core Vocabulary** terms are those that students should learn to use accurately in discussion. During instruction, expose students repeatedly to these terms but not through isolated drill or memorization.

**hardware** **reboot** **software** **technology**

**Language of Instruction** consists of additional terms that you should use when talking about concepts in the lesson. Students benefit from your modeling the use of these words without the expectation that students will use or explain the words themselves.

**component** **function** **icon** **program**

## Instructional Resource

Student Reader

Ch. 1

**Student Reader, Chapter 1**
What Is Computer Hardware?

## Materials and Equipment

**Collect or prepare the following:**

- classroom computer, tablet, or other computing device as needed for discussion

# The Core Lesson

## 1. Focus attention on today's topic.

Where can you find computers? Computers are everywhere—on our desks, in our laps, in our pockets, in our toys, and even in our kitchen appliances. People use computer devices to perform a variety of tasks for work, school, and play. Point to a computer device used in the classroom, and discuss its function and components. Ask students to name a computer device they are familiar with and how it is used.

## 2. Read together "What Is Computer Hardware?"

Student Reader

Ch. 1

Read together or have students read independently "What Is Computer Hardware?" Chapter 1 in the Student Reader. The selection introduces the idea that toys have evolved as computer technology has improved. Now we are watching Jenni get a cat that acts lifelike even though it is a piece of computer hardware.

### Guided Reading Supports

When reading aloud together as a class, always prompt students to follow along. Pause for discussion. Include suggested questions and prompts:

**Page 2**

How do you think Jenni feels in the first picture? Why?

» She looks happy with her new robot cat.

Why doesn't Jenni's mom get her a live cat?

» Jenni is allergic to cats. Having a live cat will make Jenni sneeze.

**Page 3**

What do you think Jenni is trying to get Algo to do?

» She will train him to eat, play, and sleep.

How does Jenni communicate with Algo so that Algo understands?

» Jenni uses the controller that came with Algo to communicate with Algo.

**Page 4**

How is Algo not like a real cat?

» Algo must be given instructions to do things, such as walk, play, and sleep. A real cat can walk, play, and sleep without being given instructions.

**Page 5**

What are some ways you can think of that computer technology has changed our world?

» Answers will vary but may include telephones. Now we carry them around instead of only having them in our homes.

**Page 6**

What is Jenni trying to get Algo to do in this picture?

» She programmed Algo's software with specific instructions on how to move around the obstacles to reach his food bowl.

Why does Jenni reboot Algo?

» Algo is spinning in circles and not following the given instructions.

**Page 7**

What is computer hardware?

» Hardware includes the components of a computing device that you can touch.

Can you name and describe the function of the hardware in the picture?

» Students identify and describe the function of desktop and laptop computers, smartphones, monitors, keyboards, mice, microphones, printers, etc. shown on page 7.

What does computer software do?

» Software provides instructions for specific tasks in a form that a computer can understand.

**Page 8**

Why do you think it's important to balance screen time?

» Answers will vary, but having time to do other things is important, too.

**Page 9**

Discuss ways in which older technology is still helpful today.

» Answers will vary.

## 3. Check for understanding.

Unplugged Activity

Ask students to name a computer device they have used or seen. Ask another student the main function, or use, of the computer device just named. Continue to gather the input from other students on a variety of devices.

**CHALLENGE**—Challenge students to make a list of the components within their device of choice.

**EXTEND**—Extend student learning by prompting students to create a diagram of a computer device. They should be sure to include both hardware and software components and labels.

## Know the Science

People use computer devices to perform a variety of tasks accurately and quickly. A computer device is composed of hardware and software. Hardware consists of the physical components. Software contains the instructions that direct the computer system to complete specific tasks. Problems with computer devices have different causes, and users should employ basic troubleshooting strategies to try to resolve them. This might include turning a device off and on to reboot it, closing and reopening an app, turning on speakers, or plugging in headphones.

## Know the Standards

*1A-CS-01 Select and operate appropriate software to perform a variety of tasks and recognize that users have different needs and preferences for the technology they use.* Computing technology has positively and negatively changed the way people live and work. In the past, if students wanted to read about a topic, they needed access to a library to find a book about it. Today, students can view and read information on the internet about a topic, or they can download e-books about it directly to a device. Such information may be available in more than one language and could be read to a student, allowing for great accessibility.

LESSON 1, PART 2

# What Are the Components of a Computing System?

**Driving Question:** How do programmers build computer programs?

**Overarching Unit Objective:** Model use of symbols, codes, signals, and steps in simple processes and problem-solving.

## At a Glance

### Learning Objectives

- ✓ Identify icons used to represent different applications.
- ✓ Identify components of a computing system.
- ✓ Describe the differences between software and hardware.

### Instructional Activities

- class discussion
- whole-class activity

### CSTA Standards

**Concept:** Computing Systems

**Subconcepts and Standards:**

**Devices: 1B-CS-01** Describe how internal and external parts of computing devices function to form a system. (P7.2)

**Hardware and Software: 1B-CS-02** Model how computer hardware and software work together as a system to accomplish tasks. (P4.4)

**Troubleshooting: 1B-CS-03** Determine potential solutions to solve simple hardware and software problems using common troubleshooting strategies.(P6.2)

For detailed information about the CSTA Standards, follow the links in the Online Resources Guide for this unit:

**www.coreknowledge.org/cksci-online-resources**

## Core Vocabulary and Language of Instruction

A Glossary at the end of this Teacher Guide lists definitions for Core Vocabulary and selected Language of Instruction.

**Core Vocabulary** terms are those that students should learn to use accurately in discussion. During instruction, expose students repeatedly to these terms but not through isolated drill or memorization.

**computer** **hardware** **software**

**Language of Instruction** consists of additional terms that you should use when talking about concepts in the lesson. Students benefit from your modeling the use of these words without the expectation that students will use or explain the words themselves.

**component** **function** **icon** **program**

## Materials and Equipment

**Collect or prepare the following:**

- large whiteboard
- internet access and screen projection capability
- sticky notes

Activity Page

AP 1.2

**Activity Page 1.2**

Computer Applications (AP 1.2)

## Advance Preparation

- Make sufficient copies of AP 1.2 for all students in the class.
- Draw a computer or a simple box in the center of the whiteboard.

# The Core Lesson

## 1. Focus attention on today's topic.

What are the components of a computing system? Remind students that a computing system is composed of hardware and software. Hardware components are the physical elements of a computer system. Draw a large circle on the whiteboard, and connect it to the computer/box sketch with a line. Label the circle "Hardware."

Software is what instructs the computer system to complete specific tasks. Draw another large circle labeled "Software," and connect it to the computer with a line.

## 2. Facilitate the activity.

Activity Page

AP 1.2

Pass out a copy of Activity Page 1.2 to each student. Have students circle the word part "App" in the word *Applications* in the Activity Page title. Make the connection that when students hear the word *app* in reference to a phone or a tablet, it is short for *application*. Assign or have students choose a computer icon or component to focus on. Instruct them to circle the icon or component. Prompt students to use

Unplugged Activity

Device-Based Activity

the back of the sheet to name their component, label it as hardware or software, and describe the function—using books or the internet for research. Students then use a sticky note to draw or write their component name. Students then take turns placing their sticky note in the appropriate circle on the whiteboard diagram. Lead a class discussion on the wide variety of components that can make up a computer system.

Ask students about the types of problems they have had using a computing system. What is the first thing that should be done to try to resolve the problem? *a reboot*

## 3. Check for understanding.

Unplugged Activity

Students take turns standing and reading their research to the class. Confirm they understand the purpose of their component and how it fits into a computer system.

**EXTEND**—Extend student learning by prompting students to find other computer devices around the classroom or at home and add sticky notes to the whiteboard.

## Know the Standards

*1A-CS-02 Use appropriate terminology in identifying and describing the function of common physical components of computing systems (hardware).* and *1A-CS-03 Describe basic hardware and software problems using accurate terminology.* When communicating about computing devices, it is important to describe basic hardware and software components with accurate terminology. Hardware components of a computer system can include desktop computers, laptop computers, tablet devices, monitors, keyboards, mice, speakers, and printers. A wide variety of software applications can be used to perform required tasks based on the needs of the user. Common software types include word processors, spreadsheets, presentation software, databases, drawing programs, social media, and web browsers.

LESSON 2, PART 1

# What Are Computer Networks?

**Driving Question:** How do programmers build computer programs?

**Overarching Unit Objective:** Model use of symbols, codes, signals, and steps in simple processes and problem-solving.

## At a Glance

### Learning Objectives

- ✓ Define *network*.
- ✓ Describe ways that people can connect to information and to other people through a network.
- ✓ Articulate the importance of behaving responsibly and respectfully of others online.

### Instructional Activities

- student reading
- whole-class discussion

### CSTA Standards

**Concept:** Networks and the Internet

**Subconcept and Standard:** **Cybersecurity: 1B-NI-05** Discuss real-world cybersecurity problems and how personal information can be protected. (P3.1)

For detailed information about the CSTA Standards, follow the links in the Online Resources Guide for this unit:

**www.coreknowledge.org/cksci-online-resources**

## Core Vocabulary and Language of Instruction

A Glossary at the end of this Teacher Guide lists definitions for Core Vocabulary and selected Language of Instruction.

**Core Vocabulary** terms are those that students should learn to use accurately in discussion. During instruction, expose students repeatedly to these terms but not through isolated drill or memorization.

**internet** **network** **password**

**Language of Instruction** consists of additional terms that you should use when talking about concepts in the lesson. Students benefit from your modeling the use of these words without the expectation that students will use or explain the words themselves.

**computing device** **connection** **cyberbullying** **digital citizenship**

**personal identifiable information**

## Instructional Resource

Student Reader

Ch. 2

**Student Reader, Chapter 2**
What Are Computer Networks?

## Materials and Equipment

**Collect or prepare the following:**

- classroom computer, tablet, or other computing device as needed for discussion

# The Core Lesson

## 1. Focus attention on today's topic.

Ask students to describe how the internet works. Explain that information is passed between multiple points (nodes) on a network. The internet is a network that enables people to connect with other people worldwide through many different points of connection. It allows people to connect using computers, smartphones, tablets, and a wide range of other computing devices.

Discuss the positive interactions that the internet facilitates, such as sharing ideas and finding information. The public and anonymous nature of electronic communication, however, also allows intimidating and inappropriate behavior in the form of cyberbullying. Remind students of the importance of being a good digital citizen and of keeping personal identifiable information private.

## 2. Read together "What Are Computer Networks?"

Student Reader

Ch. 2

Read together or have students read independently "What Are Computer Networks?" Chapter 2 in the Student Reader. The selection introduces the idea that we can use computer devices to connect with people on the internet. However, when we do so, we need to be mindful of what it means to be a good digital citizen.

### Guided Reading Supports

When reading aloud together as a class, always prompt students to follow along. Pause for discussion. Include suggested questions and prompts:

**Page 10**

Why is it hard for Jenni to show Algo to her grandmother?

» Grandma lives far away.

How is Jenni able to show Grandma Algo's tricks?

» Jenni's mom sets up a video call over the internet.

**Page 11**

How does Jenni send a photo of her and Algo to Grandma?

» She attaches a photo to an email and sends it to Grandma over the internet.

How does the email get the photo to Grandma?

» The email traveles over the internet, which is a network of many connection points.

**Page 12**

What does Jenni's mom remind her about sharing information over the internet?

» She tells her to never share private information or passwords with anyone.

What does Jenni's mom remind her about being a good digital citizen?

» She tells her to always be kind and respectful when communicating over the internet.

**Page 13**

What does this diagram represent?

» This diagram represents the internet, which is a network of computing devices.

Can you find a path that allows them to communicate?

» Students use their fingers to trace the network path from Jenni to Grandma.

If a path does not go through the internet, what does that mean?

» There would not be a connection to communicate with others if the path does not go through the internet.

## 3. Check for understanding.

Unplugged Activity

Ask students to explain how an email or a text message gets from one person to another. They should be able to explain that the message travels over the internet, which is a network of connection points. Ask students what it means to be a good digital citizen. They should be able to explain that it means always being kind and respectful and never sharing private information.

**CHALLENGE**—Challenge students to make a network diagram of how the computer devices in your school or another location connect to the internet.

**EXTEND**—Extend student learning by prompting students to make a list of rules for being a good digital citizen.

## Know the Science

Online Resources

View the video to learn more about how information is passed between multiple points, or nodes, on a network. See the Online Resource Guide for a link to the recommended video.

**www.coreknowledge.org/cksci-online-resources**

LESSON 2, PART 2

# What Is the Internet?

**Driving Question:** How do programmers build computer programs?

**Overarching Unit Objective:** Model use of symbols, codes, signals, and steps in simple processes and problem-solving.

## At a Glance

### Learning Objectives

- ✓ Describe how the internet allows people in different places to communicate.
- ✓ Identify physical components required for internet access.

### Instructional Activities

- class activity
- whole-class discussion

### CSTA Standards

**Concept:** Networks and the Internet

**Subconcept and Standard:** **Cybersecurity: 1B-NI-05** Discuss real-world cybersecurity problems and how personal information can be protected. (P3.1)

For detailed information about the CSTA Standards, follow the links in the Online Resources Guide for this unit:

**www.coreknowledge.org/cksci-online-resources**

## Core Vocabulary and Language of Instruction

A Glossary at the end of this Teacher Guide lists definitions for Core Vocabulary and selected Language of Instruction.

**Core Vocabulary** terms are those that students should learn to use accurately in discussion. During instruction, expose students repeatedly to these terms but not through isolated drill or memorization.

**cyberbullying** **digital citizenship** **internet** **newtork**

**Language of Instruction** consists of additional terms that you should use when talking about concepts in the lesson. Students benefit from your modeling the use of these words without the expectation that students will use or explain the words themselves.

**connection** **interaction** **personal identifiable information**

## Materials and Equipment

**Collect or prepare the following:**

- medium paper cups (2 per student)
- string (6 feet per student)
- pushpins to poke holes

## Advance Preparation

- Cut string into approximately 6-foot sections.
- Assemble one cup line for demonstration by tying both ends of the string into two cups.
- Poke holes into the bottoms of all the cups.

# The Core Lesson

## 1. Focus attention on today's topic.

What is the internet? The internet is a network that enables people to connect with other people worldwide through many different points of connection. It allows people to connect with others using computers, smartphones, tablets, and a wide range of other computing devices. Communicating through the internet allows us to share ideas, meet new people, and work together. It is important to keep all electronic communication positive and responsible.

## 2. Facilitate the activity.

Demonstrate a cup phone/cup line by asking a student to come up and put their ear on one cup while you speak softly into the other cup. Explain how the sound of your voice is sent as waves that vibrate the bottom of the cup. The vibrations are transferred to the string and into the bottom of the other cup to carry the message into the other person's ear.

Unplugged Activity

Tell students to make sure the bottoms of two paper cups have small holes and to then tie six feet of string between the two cups. With a partner, students practice using the cups to communicate—holding the string tightly and using a voice volume that cannot be heard without the use of the cup line. One partner speaks short messages into their cup line for their partner to hear with the other cup to their ear. After all pairs have confirmed both of their cup lines function as expected, the class comes back together for a whole-group activity.

Students line up where they can stand six feet apart and have one cup to speak into in one hand and another to listen to from the previous student. Begin with a short, simple, and positive message students are not familiar with, and tell the first student in the cup network (e.g., "Superstars learn in this room!"). They speak the EXACT message into their cup to the next person down the line with their ear on the cup.

The message then continues through the cup network, and the last person in line says the message out loud to see if it carried through the network clearly.

Multiple, smaller networks might be needed depending on the number of students.

## 3. Check for understanding.

Unplugged Activity

After the activity, ask students to explain what their role was in building a network. They should describe themselves as connection points or nodes in the cup network. Prompt them to explain how the message would not go through without these connection points that build a network.

Ask students why it is important to always be respectful with the messages we send through any network. Prompt them to describe how hearing an unkind or disrespectful message could make someone feel. Be sure they know to come to a trusted adult if they ever see inappropriate messages on their computing devices.

**CHALLENGE**—Break up the lines of a nursery rhyme or song students are familiar with. Challenge student teams of four or five to send these more complex messages through their cup network. The students at the end collect all the messages and try to reassemble them in the correct order to recreate the nursery rhyme or song.

**EXTEND**—Extend student learning by creating a different type of network using mirrors and flashlights blinking in a pattern for a visual network. They could also use taps in the palms of their hands to create a tactile network.

## Know the Standards

*K-2.IC.19 Work respectfully and responsibly with others when communicating electronically.* Learn more about cyberbullying and helping students learn how to work respectfully and responsibly with others when communicating electronically by using resources in the Online Resources Guide.

LESSON 3, PART 1

# What Should I Share Online?

**Driving Question:** How do programmers build computer programs?

**Overarching Unit Objective:** Model use of symbols, codes, signals, and steps in simple processes and problem-solving.

## At a Glance

### Learning Objectives

- ✓ Identify what information is safe and appropriate to share online.
- ✓ Recognize why safety is important when using computing devices and the internet.

### Instructional Activities

- student reading
- whole-class discussion

### CSTA Standards

**Concept:** Networks and the Internet

**Subconcepts and Standard:** **Cybersecurity: 1B-NI-05** Discuss real-world cybersecurity problems and how personal information can be protected. (P3.1)

For detailed information about the CSTA Standards, follow the links in the Online Resources Guide for this unit:

**www.coreknowledge.org/cksci-online-resources**

## Core Vocabulary and Language of Instruction

A Glossary at the end of this Teacher Guide lists definitions for Core Vocabulary and selected Language of Instruction.

**Core Vocabulary** terms are those that students should learn to use accurately in discussion. During instruction, expose students repeatedly to these terms but not through isolated drill or memorization.

**digital footprint** **log off** **online** **password** **username**

**Language of Instruction** consists of additional terms that you should use when talking about concepts in the lesson. Students benefit from your modeling the use of these words without the expectation that students will use or explain the words themselves.

**privacy** **protection** **safety** **security** **unauthorized**

## Instructional Resource

Student Reader

Ch. 3

**Student Reader, Chapter 3**
What Should I Share Online?

## Materials and Equipment

**Collect or prepare the following:**

- classroom computer, tablet, or other computer device as needed for discussion

# The Core Lesson

## 1. Focus attention on today's topic.

What should I share online? Lead a discussion about staying safe when using the internet. Ask students to share examples of private and public information. Ask them to decide which is okay to share online and which isn't. Define a digital footprint as all the information that exists about you on the internet, and explain that it might be permanently accessible. Explain that passwords are like a first line of defense to protect your personal information online. Ask students if they would ever share a password with anyone except their parents.

## 2. Read together "What Should I Share Online?"

Student Reader

Ch. 3

Read together or have students read independently "What Should I Share Online?" Chapter 3 in the Student Reader. The selection introduces the idea that when you are using the internet, you must always be safe and aware that what you say and do is left behind for others to see.

### Guided Reading Supports

When reading aloud together as a class, always prompt students to follow along. Pause for discussion. Include suggested questions and prompts:

**Page 14**

Why do you think Jenni's mom is concerned about her joining the RoboCat community?

» She knows that sometimes people post unkind messages and can be mean or rude.

What does Jenni need to promise?

» She has to promise that she will not post unkind messages and to show her mom if anyone is ever mean or rude online.

**Page 15**

What are some of the things Jenni shares online with her friends?

» She shares ideas for new tricks, photos, and ways to solve problems.

Why is it important to never share personal information online?

» Everything you share online creates a digital footprint and may always be accessible on the internet.

**Page 16**

What does Jenni keep private when she is online?

» her name and login information

Why does Jenni create a strong password?

» She creates a strong password so no one else can access her account information.

Why does Jenni log out of her account when she is finished?

» She always logs out so nobody else can access her account.

**Page 17**

Who can name one rule for creating a strong password?

» Students name anything from the list in their Student Reader.

Who can find a strong password for Jenni on page 17? Please explain why it is so strong.

» Students identify the strong passwords and explain why they meet all the criteria.

## 3. Check for understanding.

Unplugged Activity

Ask students to raise their hands to share what staying safe online means to them. Tell them to also name one example of private information that should not be shared online. Prompt students to reflect on their own passwords and if they meet the criteria of a strong password.

**CHALLENGE**—Challenge students to come up with as many of the easiest passwords they can think of (e.g., 12345, ABCDEF, myname, password, etc.).

**EXTEND**—Extend student learning by creating a class poster about ways to stay safe online.

LESSON 3, PART 2

# What Is a Digital Footprint?

**Driving Question:** How do programmers build computer programs?

**Overarching Unit Objective:** Model use of symbols, codes, signals, and steps in simple processes and problem-solving.

## At a Glance

### Learning Objectives

- ✓ Explain the concept of a digital footprint.
- ✓ Understand that login information should be kept private.

### Instructional Activities

- small-group activity
- whole-class discussion

### CSTA Standards

**Concept:** Networks and the Internet

**Subconcept and Standard:** **Cybersecurity: 1B-NI-05** Discuss real-world cybersecurity problems and how personal information can be protected. (P3.1)

For detailed information about the CSTA Standards, follow the links in the Online Resources Guide for this unit:

**www.coreknowledge.org/cksci-online-resources**

## Core Vocabulary and Language of Instruction

A Glossary at the end of this Teacher Guide lists definitions for Core Vocabulary and selected Language of Instruction.

**Core Vocabulary** terms are those that students should learn to use accurately in discussion. During instruction, expose students repeatedly to these terms but not through isolated drill or memorization.

**digital footprint** **log off** **password** **permanent** **privacy**

**Language of Instruction** consists of additional terms that you should use when talking about concepts in the lesson. Students benefit from your modeling the use of these words without the expectation that students will use or explain the words themselves.

**protection** **safety** **security** **unauthorized**

## Materials and Equipment

**Collect or prepare the following:**

- large whiteboard
- internet access and screen projection capability
- Printed copies of the Digital Trails—Animal Tracks Squares
- printed copies of the Digital Trails—Animal Tracks student handout

## Advance Preparation

Online Resources

- Cue up the video "Follow the Digital Trail" for whole-class viewing.
- Print and cut out the Digital Trails—Animal Tracks Squares.
- Print 4–5 copies of the Digital Trails—Animal Tracks student handout for each group.

See the Online Resource Guide for links to these resources.

**www.coreknowledge.org/cksci-online-resources**

- Prepare a chart with two columns. Label the columns "Okay to Share" and "NOT Okay to Share"

# The Core Lesson

## 1. Focus attention on today's topic.

Online Resources

What is a digital footprint? Discuss the difference between public and private information. Ask students makes something private, and ask why they would not want others to be able to access that information. Remind them that a strong password is like a first line of defense to protecting private information online.

Play the short video "Follow the Digital Trail" for the class.

Invite students to share their explanation of a digital footprint. Clarify that a digital footprint is a record (or trail) of all the information about you shared online and all the sites you visit. This is why it is important to be careful not to share any private information online, such as your full name, address, phone number, age, or birthday, where strangers might be able to find it.

## 2. Facilitate the activity.

Online Resources

Unplugged Activity

Place the cut-out Digital Trails—Animal Tracks Squares on the ground in two different trails, keeping Mervin the Mouse's and Ellie the Elephant's trails separate from each other.

Tell the class that Mervin the Mouse and Ellie the Elephant were playing a game online and noticed a chat box. They thought it would be fun to talk to the other players and started to share some information about themselves. The only problem is that they forgot to ask their parents if it was okay first.

Place students into groups of four, and distribute the Animal Tracks student handout to each group. Tell students that the information that Mervin and Ellie posted can be seen by anyone online.

Groups take turns following the Animal Tracks for Mervin and for Ellie. They fill out what information they find for each animal on their student handout as they follow their trails. Groups also complete the two questions on the bottom of their handout:

Ask students who they found more about and why?

- Explain that the answer is Ellie, since they now know where she lives, what she looks like, and a lot of private information about her life.

Ask students if there is anything that Ellie or Mervin posted that could be a problem for them.

- Explain that while it is okay to share fun and interesting information online, such as nicknames or hobbies, sharing private information like their address, their full name, and their school name is not safe. Remind them that sharing secrets or unkind things online can hurt other people's feelings.
- Explain that it's important to think carefully about what we share online because the information is permanent—it lasts forever!

(Adapted from: https://studio.code.org/s/courseb-2022/lessons/1)

## 3. Check for understanding.

Project the chart, and add student responses to which information is okay and which is not okay to share online. Suggested responses:

Unplugged Activity

| Okay to Share | NOT Okay to Share |
|---|---|
| Interests | Full name |
| Hobbies | Address |
| Usernames | Birth date |
| | Passwords |
| | School name |
| | Information that would hurt others |

## Know the Standards

Online Resources

*1A-NI-04 Explain what passwords are and why we use them and use strong passwords to protect devices and information from unauthorized access.* and *1A-IC-18 Keep login information private and log off of devices appropriately.* It is important to focus on cybersecurity even in the younger grades so students make protecting themselves online a habit with strong passwords, understanding private versus public information, protecting one's digital footprint, and logging off of public devices. See the Online Resources Guide for a link to learn more about digital footprints.

**www.coreknowledge.org/cksci-online-resources**

Emphasize that students should ask their parents or caregivers for permission before sharing any information about themselves online.

**CHALLENGE**—Challenge students to create a Digital Trail for their own character. They should include things that are okay and not okay to share online.

**EXTEND**—Extend student learning by having each student complete the last page of the Animal Tracks student handout. They draw a picture of something that is okay to share about themselves on the internet and write one thing that is not okay to share on the internet.

LESSON 4, PART 1

# What Is Digital Data?

**Driving Question:** How do programmers build computer programs?

**Overarching Unit Objective:** Model use of symbols, codes, signals, and steps in simple processes and problem-solving.

## At a Glance

### Learning Objectives

- ✓ Define the terms *binary code, binary digit,* and *bit.*
- ✓ Recognize that computers use binary numbers to represent information.

### Instructional Activities

- student reading
- whole-class discussion

### CSTA Standards

**Concept:** Data Analysis

**Subconcept and Standard: Storage: 1A-DA-05** Store, copy, search, retrieve, modify, and delete information using a computing device and define the information stored as data. (P4.2)

For detailed information about the CSTA Standards, follow the links in the Online Resources Guide for this unit:

**www.coreknowledge.org/cksci-online-resources**

## Core Vocabulary and Language of Instruction

A Glossary at the end of this Teacher Guide lists definitions for Core Vocabulary and selected Language of Instruction.

**Core Vocabulary** terms are those that students should learn to use accurately in discussion. During instruction, expose students repeatedly to these terms but not through isolated drill or memorization.

**bit** **binary** **binary code** **code** **digital data**

**Language of Instruction** consists of additional terms that you should use when talking about concepts in the lesson. Students benefit from your modeling the use of these words without the expectation that students will use or explain the words themselves.

**processing** **storage**

## Instructional Resource

Student Reader

Ch. 4

**Student Reader, Chapter 4**
What Is Digital Data?

## Materials and Equipment

**Collect or prepare the following:**

- classroom computer, tablet, or other computer device as needed for discussion

# The Core Lesson

## 1. Focus attention on today's topic.

What is digital data? Explain that computers can only work with digital information. Everything that gets processed by a computer must first be turned into a digital signal. To process information and store it, computers use binary code. Binary is a language composed of just the numbers 1 and 0, which represent "on" and "off."

## 2. Read together "What Is Digital Data?"

Student Reader

Ch. 4

Read together or have students read independently "What Is Digital Data?" Chapter 4 in the Student Reader. The selection introduces the idea that a computer uses binary code in order to work. In this selection, the binary code is described using "on" and "off" features of the robot cat.

### Guided Reading Supports

When reading aloud together as a class, always prompt students to follow along. Pause for discussion. Include suggested questions and prompts:

**Page 18**

What times of day does Jenni feed Algo?

» She feeds him at seven in the morning and five in the evening.

How does she tell Algo to eat at those times?

» She has programmed Algo's feeding schedule to be ON at those times and OFF all the other times.

**Page 19**

What hours is Algo set to sleep?

» 8–11 p.m. and 12–6 a.m.

What hours is Algo set to play?

» 8 a.m. and 3 p.m.

What hours is Algo set to meow?

» Algo is not set to meow.

What activities are set to ON for Algo?

» sleep, feed, play

**Page 20**

Algo's feeding schedule is an example of binary data, which is information that can be one of two values. In Algo's case, his feeding hours can only be set to ON or OFF. Binary data make up the code, or the instructions to a computing device on what tasks to do and what information to store.

Why is Algo's feeding schedule binary data?

» It can only have two values—ON or OFF.

**Page 21**

How is binary code built?

» It is built up from bits.

How do bits become meaningful?

» Bits are combined in certain patterns that make up numbers, letters, and symbols.

## 3. Check for understanding.

Ask students to give an example of something that is binary—something with only two states. Tell them to describe the ON state with a 1 and the OFF state with a 0.

Unplugged Activity

**CHALLENGE**—Challenge students to find other things in their world that are binary—something with two values (e.g., light or dark, night or day, up or down, in or out, etc.).

**EXTEND**—Extend student learning by asking students to draw a clock or a schedule and highlight the times when their sleeping or eating schedules are turned ON.

## Know the Science

Online Resources

Consider binary as an *abstraction*—a way of simplifying the complexities of the inner workings of a computing device that processes and stores data. See the Online Resource Guide for a link to learn more about how binary digits work.

**www.coreknowledge.org/cksci-online-resources**

LESSON 4, PART 2

# How Can 1s and 0s Make Letters?

**Driving Question:** How do programmers build computer programs?

**Overarching Unit Objective:** Model use of symbols, codes, signals, and steps in simple processes and problem-solving.

## At a Glance

### Learning Objective

✓ Encode letters into binary data, and decode binary data back to letters.

### Instructional Activities

- class activity
- whole-class discussion
- paired challenge

### CSTA Standards

**Concept:** Data Analysis

**Subconcept and Standard:** **Storage: 1A-DA-05** Store, copy, search, retrieve, modify, and delete information using a computing device, and define the information stored as data. (P4.2)

For detailed information about the CSTA Standards, follow the links in the Online Resources Guide for this unit:

**www.coreknowledge.org/cksci-online-resources**

## Core Vocabulary and Language of Instruction

A Glossary at the end of this Teacher Guide lists definitions for Core Vocabulary and selected Language of Instruction.

**Core Vocabulary** terms are those that students should learn to use accurately in discussion. During instruction, expose students repeatedly to these terms but not through isolated drill or memorization.

**binary** **bit** **byte** **data** **digital**

**Language of Instruction** consists of additional terms that you should use when talking about concepts in the lesson. Students benefit from your modeling the use of these words without the expectation that students will use or explain the words themselves.

**code** **processing** **storage**

## Materials and Equipment

**Collect or prepare the following:**

- large whiteboard
- internet access and screen projection capability
- blank paper or student notebooks
- Binary Decoder Key

See the Online Resources Guide for a link to this item.

**www.coreknowledge.org/cksci-online-resources**

## Advance Preparation

Online Resources

- Cue up the Binary Decoder Key with the projector.
- Using the Binary Decoder Key, create a simple three-letter word on the whiteboard using the appropriate binary symbols.
- Create a longer but simple word on the whiteboard, but cover it until the end of the lesson.

# The Core Lesson

## 1. Focus attention on today's topic.

How can 1s and 0s make letters? Discuss how computers can only process information that is created with binary code. Binary is a language composed of just the numbers 1 and 0, which represent "on" or "off." Computer programmers use combinations of 1s and 0s to represent every letter of the alphabet, every number, every color, etc.

## 2. Facilitate the activity.

Online Resources

Project the Binary Decoder Key for the class, and discuss how the key shows how each block pattern represents a letter. Each letter of the alphabet has a combination of 1s and 0s, or ON and OFF, that the computer can read.

Point to the three-letter word you created on the whiteboard. Work with the class to find each pattern in the key and determine which letter it represents. Write the three letters on the board, and ask students what word was made.

Next, ask students to name another three- or four-letter word. Write the word vertically on the board, and then work with students to draw the proper binary code pattern to represent the word. If students are clear on the concept, create the word with 1s and 0s instead of the dark or light squares.

Keep the Binary Decoder Key projected, and prompt students to use a blank sheet of paper to create their initials in binary code—without revealing the letters they are representing. Collect all of the papers, shuffle, and pass them out around the room. Students decode the initials on the sheet they are given. When everyone is ready, students take turns sharing the initials they decoded and guessing whom they belong to.

## 3. Check for understanding.

Unplugged Activity

Reveal one final, longer word or name that you have created on the whiteboard. Students work with a partner to decode and write down their answer for you to check. Student pairs can volunteer to share what they decoded with the class.

**CHALLENGE**—Challenge students to visualize binary code as you read a series of 1s and 0s instead of them seeing it on the board. Spell out three or more letters, and ask them to use the projected Binary Decoder Key to find the word.

**EXTEND**—Extend student learning by asking student pairs to write and decipher simple words or messages to each other in binary code using 1s and 0s.

## Know the Standards

Online Resources

*1A-DA-05 Store, copy, search, retrieve, modify, and delete information using a computing device and define the information stored as data.* All information stored and processed by a computing device is referred to as data. Data can be words, numbers, images, text documents, audio files, software programs or apps, video files, etc., which are all just long patterns of 1s and 0s. See the Online Resources Guide for a link to a video to learn more about binary code and data.

**www.coreknowledge.org/cksci-online-resources**

LESSON 5, PART 1

# How Can Data Be Organized?

**Driving Question:** How do programmers build computer programs?

**Overarching Unit Objective:** Model use of symbols, codes, signals, and steps in simple processes and problem-solving.

## At a Glance

### Learning Objective

✓ Classify items and information by characteristics.

### Instructional Activities

- student reading
- whole-class discussion

### CSTA Standards

**Concept:** Data Analysis

**Subconcepts and Standards:**

**Collection, Visualization, and Transformation: 1B-DA-06** Organize and present collected data visually to highlight relationships and support a claim. (P7.1)

**Inference and Models: 1B-DA-07** Use data to highlight or propose cause-and-effect relationships, predict outcomes, or communicate an idea. (P7.1)

For detailed information about the CSTA Standards, follow the links in the Online Resources Guide for this unit:

**www.coreknowledge.org/cksci-online-resources**

## Core Vocabulary and Language of Instruction

A Glossary at the end of this Teacher Guide lists definitions for Core Vocabulary and selected Language of Instruction.

**Core Vocabulary** terms are those that students should learn to use accurately in discussion. During instruction, expose students repeatedly to these terms but not through isolated drill or memorization.

**data** **graph** **pattern** **update** **visualize**

**Language of Instruction** consists of additional terms that you should use when talking about concepts in the lesson. Students benefit from your modeling the use of these words without the expectation that students will use or explain the words themselves.

**attribute** **characteristic** **prediction** **visualization**

## Instructional Resource

Student Reader

Ch. 5

**Student Reader, Chapter 5**
How Can Data Be Organized?

## Materials and Equipment

**Collect or prepare the following:**

- classroom computer, tablet, or other computer device as needed for discussion

# The Core Lesson

## 1. Focus attention on today's topic.

What is data? Remind students that data is information that is stored and processed by a computing device. Explain that the collection and display of data is important in our lives. Data can be collected and organized by sorting information into groups and visualized in a variety of formats, including charts and graphs. Ask students if they see any charts or graphs in the room, such as a job chart. Data can be used to learn and make predictions about the world.

## 2. Read together "How Can Data Be Organized?"

Student Reader

Ch. 5

Read together or have students read independently "How Can Data Be Organized?" Chapter 5 in the Student Reader. The selection introduces the idea that computers collect data and display it in ways that are easy for the user to understand.

### Guided Reading Supports

When reading aloud together as a class, always prompt students to follow along. Pause for discussion. Include suggested questions and prompts:

**Page 22**

Why does Jenni plug in Algo?

» She wants to recharge his battery and also exchange information with the computer.

What information does the program on the computer collect?

» It keeps track of how Algo has been working and also sends new instructions, or updates, to Algo.

**Page 23**

Why is Algo's data easy for Jenni to understand?

» Data is not just a list of numbers; it is represented in colorful charts and graphs.

**Page 24**

What does Jenni's mom have that collects data like Algo's computer program?

» Her mom has a fitness tracker that she wears around her wrist.

What do both Mom's and Algo's data tell them?

» The data shows how much they sleep during the night and when they wake up.

**Page 25**

Why is Jenni looking at Algo's data?

» Jenni wants to be sure Algo has the right amounts of sleep, food, playtime, and cuddling.

From the graph of Algo's data, what makes him happiest?

» playing and cuddling

## 3. Check for understanding.

**Unplugged Activity**

Ask students, "In what ways did you see the data from Algo being organized?" Guide students to recognize that his sleep patterns and happiness level were put into different types of charts. Explain that when things are displayed in a way that is easy for the user to understand, the user is able to learn from the data they are collecting. Tell students to turn and talk with a partner about ways they have seen computer devices collect and display data.

**EXTEND**—Extend student learning by asking students to collect data about their classmates' pets and come up with a visual way to represent the data.

## Know the Science

**Online Resources**

Graphs and charts condense large amounts of information into formats that are quicker and easier to understand and help us make more effective use of that data. See the Online Resource Guide for a link to learning resources about choosing the right visual display for data.

**www.coreknowledge.org/cksci-online-resources**

LESSON 5, PART 2

# How Can Data Be Used to Make Predictions?

**Driving Question:** How do programmers build computer programs?

**Overarching Unit Objective:** Model use of symbols, codes, signals, and steps in simple processes and problem-solving.

## At a Glance

### Learning Objectives

- ✓ Collect and categorize data in up to three categories.
- ✓ Display data in a visual format.
- ✓ Identify patterns in data to make a prediction.

### Instructional Activities

- whole-class discussion
- small-group activity
- class activity

### CSTA Standards

**Concept:** Data Analysis

**Subconcepts and Standards:**

**Collection, Visualization, and Transformation: 1B-DA-06** Organize and present collected data visually to highlight relationships and support a claim. (P7.1)

**Inference and Models: 1B-DA-07** Use data to highlight or propose cause-and-effect relationships, predict outcomes, or communicate an idea. (P7.1)

For detailed information about the CSTA Standards, follow the links in the Online Resources Guide for this unit:

**www.coreknowledge.org/cksci-online-resources**

## Core Vocabulary and Language of Instruction

A Glossary at the end of this Teacher Guide lists definitions for Core Vocabulary and selected Language of Instruction.

**Core Vocabulary** terms are those that students should learn to use accurately in discussion. During instruction, expose students repeatedly to these terms but not through isolated drill or memorization.

**collect** **data** **graph** **pattern** **predict** **visualize** **update**

**Language of Instruction** consists of additional terms that you should use when talking about concepts in the lesson. Students benefit from your modeling the use of these words without the expectation that students will use or explain the words themselves.

**attribute** **characteristic** **prediction** **visualization**

## Materials and Equipment

**Collect or prepare the following:**

- large whiteboard
- a minimum of 5 small bags of small, colored candy
- blank paper

## Advance Preparation

- Clear the whiteboard.
- Purchase bags of candy.

# The Core Lesson

## 1. Focus attention on today's topic.

How can data be used to make predictions? Data can be collected from the real world and then processed and stored by computing devices. Data can be organized and presented in formats, such as charts and graphs, that help us see patterns, communicate clearly, and make predictions about the world.

## 2. Facilitate the activity.

Online Resources

Unplugged Activity

Using a common candy with multiple colors, students will work in small groups to make a chart and a graph representing the quantity of each candy color within a bag.

Place students into four or five small groups, and provide each group with a bag of the same type of candy. Be sure to save one bag of candy for the whole-class activity at the end. Groups count how many colors they see in the bag and make that number of columns on a blank sheet of paper. They then sort the candy colors into the different columns and label each column with the quantity of that color. The groups discuss which color was most common, least common, and so on.

On the whiteboard, student groups each create a bar graph to represent their data. Assist them in determining how many bars to make and the scale.

After all groups have created their graphs, lead a whole-class discussion on any patterns they see across all of the bags. Did all bags have the same colors represented? Was the greatest quantity always from the same color? Which colors seem to have the least quantity across the classroom?

Remind students that there is one more bag of candy to analyze. What predictions can they make about its contents? Invite students to predict the following:

- Will the same set of colors be represented?
- Which color will have the greatest quantity?
- Which color will have the least quantity?

Ask students to share their predictions and why they think they will occur.

## 3. Check for understanding.

Unplugged Activity

Pull out each piece from the bag, and have students take turns adding the data to a bar graph on the board. Were their predictions accurate? What other observations can they make about the class bag? Was anything surprising when comparing their group's data with other groups' data or with the data from the class bag?

**SUPPORT**—Small manipulatives or erasers could be placed in bags randomly for student groups to complete the activity.

**CHALLENGE**—Challenge students to graph a larger set of data using a larger bag of colored candy. They can then create a bar graph to represent their findings.

**ALERT**—Be sure any purchased candy or food items meet the classroom and school policies for food allergies.

## Know the Standards

*1A-DA-06 Collect and present the same data in various visual formats.* and *1A-DA-07 Identify and describe patterns in data visualizations, such as charts or graphs, to make predictions.* The collection and use of data about the world around us is a routine part of life and influences how we live. Helping students identify and describe patterns in their world is enhanced by the use of charts or graphs. Identifying patterns helps us make predictions based on the patterns observed, such as with weather data.

LESSON 6, PART 1

# Symbols to Communicate

**Driving Question:** How do programmers build computer programs?

**Overarching Unit Objective:** Model use of symbols, codes, signals, and steps in simple processes and problem-solving.

## At a Glance

### Learning Objective

✓ Identify ways in which information can be represented by alphanumeric symbols and pictures.

### Instructional Activities

- student reading
- whole-class discussion

### CSTA Standards

**Concept:** Algorithms and Programming

**Subconcept and Standard: Variables: 1B-AP-09** Create programs that use variables to store and modify data. (P5.2)

For detailed information about the CSTA Standards, follow the links in the Online Resources Guide for this unit:

**www.coreknowledge.org/cksci-online-resources**

## Core Vocabulary and Language of Instruction

A Glossary at the end of this Teacher Guide lists definitions for Core Vocabulary and selected Language of Instruction.

**Core Vocabulary** terms are those that students should learn to use accurately in discussion. During instruction, expose students repeatedly to these terms but not through isolated drill or memorization.

**communicate** **computer code** **language** **symbol**

**Language of Instruction** consists of additional terms that you should use when talking about concepts in the lesson. Students benefit from your modeling the use of these words without the expectation that students will use or explain the words themselves.

**direction** **represent** **sequence**

## Instructional Resource

Student Reader

Ch. 6

**Student Reader, Chapter 6**

Symbols to Communicate

## Materials and Equipment

**Collect or prepare the following:**

- classroom computer, tablet, or other computer device for discussion

# The Core Lesson

## 1. Focus attention on today's topic.

How can you describe things without using words? Describe how information in the real world can be represented using symbols. For example, a thumbs up or thumbs down can mean YES or NO. Arrows can represent directions. Symbols can help humans communicate with computer devices, which do not have the ability to produce or understand human language.

## 2. Read together "Symbols to Communicate"

Student Reader

Ch. 6

Read together or have students read independently "Symbols to Communicate" Chapter 6 in the Student Reader. The selection introduces the idea that communication can happen even if words aren't being used. Using pictorial representations of objects allows people to communicate with one another, whether it be in person or when using computers.

### Guided Reading Supports

When reading aloud together as a class, always prompt students to follow along. Pause for discussion. Include suggested questions and prompts:

**Page 26**

What are some of the ways humans and pets communicate?

» Humans give pets instructions using language or their voice. Pets express their needs and feelings to humans by barking, meowing, purring, or hissing.

How does Algo understand instructions?

» Algo is programmed to follow instructions using computer code.

**Page 27**

How do Jenni and Algo tell each other things?

» They use symbols to represent needs and wants.

What symbols let Jenni know Algo wants to eat or play?

» Algo shows a bowl symbol when he wants to eat and a ball symbol when he wants to play.

**Page 28**

What type of symbol helps Jenni tell Algo how to move?

» Jenni uses arrows to tell Algo where to move.

What arrows would Jenni press to tell Algo to move forward and then to the right?

» Jenni would press the up arrow to tell Algo to move forward and then the right arrow to move him right.

**Page 29**

What needs does Algo communicate with these symbols?

» water bowl (thirsty)
» petting (love)
» sleeping (tired)
» ball of yarn (playtime)
» food bowl (hungry)
» litter box (bathroom)
» licking paw (cleaning)

## 3. Check for understanding.

**Unplugged Activity**

Ask students to describe how a communication between Jenni and Algo might take place. If necessary, prompt them with a need, such as "thirsty" or "tired." Students should describe what the need is, how Algo or Jenni communicates it, and the resulting action.

**CHALLENGE**—Challenge students by asking them to invent and draw other symbols that Jenni and Algo could use to communicate.

**EXTEND**—Extend student learning by asking students to make a list of or draw ten symbols they see in their daily lives and what they represent, such as a wheelchair representing a parking space for a person with a disability or a cart representing a shopping cart on an online store.

LESSON 6, PART 2

# What Does the Symbol Mean?

**Driving Question:** How do programmers build computer programs?

**Overarching Unit Objective:** Model use of symbols, codes, signals, and steps in simple processes and problem-solving.

## At a Glance

### Learning Objectives

- ✓ Define *symbol*.
- ✓ Model the way programs store and manipulate data.

### Instructional Activities

- small-group activity
- whole-class discussion

### CSTA Standards

**Concept:** Algorithms and Programming

**Subconcept and Standard: Variables: 1B-AP-09** Create programs that use variables to store and modify data. (P5.2)

For detailed information about the CSTA Standards, follow the links in the Online Resources Guide for this unit:

**www.coreknowledge.org/cksci-online-resources**

## Core Vocabulary and Language of Instruction

A Glossary at the end of this Teacher Guide lists definitions for Core Vocabulary and selected Language of Instruction.

**Core Vocabulary** terms are those that students should learn to use accurately in discussion. During instruction, expose students repeatedly to these terms but not through isolated drill or memorization.

**communicate** **language** **symbol**

**Language of Instruction** consists of additional terms that you should use when talking about concepts in the lesson. Students benefit from your modeling the use of these words without the expectation that students will use or explain the words themselves.

**direction** **represent**

## Materials and Equipment

**Collect or prepare the following:**

- internet access and screen projection capability
- blank paper and pencil

## Advance Preparation

- Scan the classroom and the school grounds to become aware of symbols available for students to find on their hunt.
- Cue up the Computerhope webpage on the projector.

See the Online Resource Guide for a link to this resource.

**www.coreknowledge.org/cksci-online-resources**

# The Core Lesson

## 1. Focus attention on today's topic.

What does the word *symbol* mean? Ask students to share their definition of a symbol with the class. Discuss how the world is full of symbols and how good symbols help us understand information more quickly and easily.

## 2. Facilitate the activity.

**Online Resources**

Place students into groups of three or four, and instruct them to go on a symbol hunt around the classroom or school. They should try to find at least five different symbols, such as a power button on a computer, a restroom sign, a recycling symbol, a speaker icon, a bike parking sign, etc. On a sheet of paper, students draw, trace, or describe the symbols they find and write what they represent.

**Device-Based Activity**

Encourage students to notice which symbols look like the item they represent (a bike parking sign or a headphone plug) and which symbols have come to represent something else (a power button or recycling symbol). Student groups can then share their findings with the rest of the class.

Next, using a projector, zoom in on some of the common computer icons on the displayed webpage. Ask students to raise their hands and share the meaning of each computer symbol and whether its intention is clear.

## 3. Check for understanding.

Unplugged Activity

To close the lesson, remind students that a symbol is a visual that represents something else. Ask students to circle the item on their symbol hunt page that they think is the clearest symbol to understand its meaning. Time permitting, have students share their favorite symbol with the class and why they feel it is an example of a clear way to represent something else.

**CHALLENGE**—Challenge students to produce nonvisual symbols, such as a gesture (a handshake or a wave) or a sound (the tardy bell or a ringtone).

**EXTEND**—Extend student learning by inviting student pairs to invent and draw their own visual symbols. Have them show other student pairs and try to guess the meaning of each other's symbols.

## Know the Standards

*1A-AP-09 Model the way programs store and manipulate data by using numbers or other symbols to represent information.* Information in the real world is represented by data in computer programs. Symbols, numbers, and pictographs are common ways that digital information displayed to help people make sense of the world around them.

LESSON 7, PART 1

# Computer Devices Follow Steps

**Driving Question:** How do programmers build computer programs?

**Overarching Unit Objective:** Model use of symbols, codes, signals, and steps in simple processes and problem-solving.

## At a Glance

### Learning Objective

✓ Define *algorithm* as step-by-step instructions used to complete a task.

### Instructional Activities

- student reading
- whole-class discussion

### CSTA Standards

**Concept:** Algorithms and Programming

**Subconcepts and Standards:**

**Algorithms: 1B-AP-08** Compare and refine multiple algorithms for the same task and determine which is the most appropriate. (P6.3, P3.3)

**Modularity: 1B-AP-11** Decompose (break down) problems into smaller, manageable subproblems to facilitate the program development process. (P3.2)

For detailed information about the CSTA Standards, follow the links in the Online Resources Guide for this unit:

**www.coreknowledge.org/cksci-online-resources**

## Core Vocabulary and Language of Instruction

A Glossary at the end of this Teacher Guide lists definitions for Core Vocabulary and selected Language of Instruction.

**Core Vocabulary** terms are those that students should learn to use accurately in discussion. During instruction, expose students repeatedly to these terms but not through isolated drill or memorization.

**algorithm** **routine** **step-by-step**

**Language of Instruction** consists of additional terms that you should use when talking about concepts in the lesson. Students benefit from your modeling the use of these words without the expectation that students will use or explain the words themselves.

**end result** **task**

## Instructional Resource

Student Reader

Ch. 7

**Student Reader, Chapter 7**

Computer Devices Follow Steps

## Materials and Equipment

**Collect or prepare the following:**

- classroom computer, tablet, or other computer device for discussion

# The Core Lesson

## 1. Focus attention on today's topic.

What are algorithms? Algorithms are step-by-step instructions that programmers "code" into computing devices to get them to complete specific tasks. Draw comparisons to step-by-step routines students do daily at school, such as getting to school, cleaning up before recess, calendar time, or packing up for the day. Highlight that each task requires individual steps that lead to an end result (arriving at school, a clean room, an up-to-date calendar, or exiting school).

## 2. Read together "Computer Devices Follow Steps" Chapter 7.

Student Reader

Ch. 7

Read together or have students read independently "Computer Devices Follow Steps" Chapter 7 in the Student Reader. The selection introduces the idea that in order to complete a task, there is a series of steps that must be followed. Sometimes those steps must be in a particular order to achieve the desired result.

### Guided Reading Supports

When reading aloud together as a class, always prompt students to follow along. Pause for discussion. Include suggested questions and prompts:

**Page 30**

What is an algorithm?

» An algorithm is a set of steps or rules that leads to a result.

When does Jenni use algorithms in her daily life?

» She uses algorithms when she gets ready for school or puts on her socks and shoes.

**Page 31**

How does Jenni use algorithms with Algo?

» She uses algorithms to tell Algo what tasks to perform.

**Page 32**

What symbols does Jenni use to instruct Algo to walk to a location?

» She uses the arrows, or direction symbols.

What does Jenni program Algo to do when his sensors detect a digital "treat"?

» She programs him to stop.

**Page 33**

Use your finger to trace a path Jenni could use to get Algo through the maze. An X represents something he must go around, and an S represents a snack. Remember that Algo cannot move backward.

» Students point out the path from Algo to his bed moving around obstacles and stopping for snacks.

## 3. Check for understanding.

Ask students to raise their hands and describe a task they perform each day at home or school in a step-by-step manner. Examples could include feeding a pet, brushing teeth, pouring a bowl of cereal, reading a book, and so on.

Unplugged Activity

**CHALLENGE**—Challenge students to create a larger grid for Algo to navigate. They can add more treats and block off squares for additional challenges.

**EXTEND**—Extend student learning by creating a similar grid on the ground with tape or outside on a blacktop with chalk. Students guide each other through the grid, stopping to pick up treats along the way.

## Know the Science

Online Resources

Algorithmic thinking is the ability to define discrete steps to complete a task or solve a problem. It is beneficial in all subject areas but especially in math and science. See the Online Resource Guide for a link to read more about the benefits of algorithmic thinking.

**www.coreknowledge.org/cksci-online-resources**

LESSON 7, PART 2

# How Do You Make an Algorithm?

**Driving Question:** How do programmers build computer programs?

**Overarching Unit Objective:** Model use of symbols, codes, signals, and steps in simple processes and problem-solving.

## At a Glance

### Learning Objectives

- ✓ Identify a real-world algorithm.
- ✓ Construct an algorithm to complete a task.

### Instructional Activities

- whole-class discussion
- individual student activity
- paired activity

### CSTA Standards

**Concept:** Algorithms and Programming

**Subconcepts and Standards:**

**Algorithms: 1B-AP-08** Compare and refine multiple algorithms for the same task and determine which is the most appropriate. (P6.3, P3.3)

**Modularity: 1B-AP-11** Decompose (break down) problems into smaller, manageable subproblems to facilitate the program development process. (P3.2)

For detailed information about the CSTA Standards, follow the links in the Online Resources Guide for this unit:

**www.coreknowledge.org/cksci-online-resources**

## Core Vocabulary and Language of Instruction

A Glossary at the end of this Teacher Guide lists definitions for Core Vocabulary and selected Language of Instruction.

**Core Vocabulary** terms are those that students should learn to use accurately in discussion. During instruction, expose students repeatedly to these terms but not through isolated drill or memorization.

**algorithm** **decomposition** **routine** **step-by-step**

**Language of Instruction** consists of additional terms that you should use when talking about concepts in the lesson. Students benefit from your modeling the use of these words without the expectation that students will use or explain the words themselves.

**composition** **end result** **task**

## Materials and Equipment

**Collect or prepare the following:**

Online Resources

- internet access and screen projection capability
- printed copies of Hello Ruby Map Algorithms (1 per student)

See the Online Reources Guide for a link to this item.

**www.coreknowledge.org/cksci-online-resources**

## Advance Preparation

- Prepare the projection of Ruby's map.

# The Core Lesson

## 1. Focus attention on today's topic.

How do you make an algorithm? Invite students to share their definition of an algorithm. Reinforce the concept of an algorithm as a step-by-step process that leads to a specific end result. Ask students why algorithms are necessary to get computing devices to do the things we want them to do.

## 2. Facilitate the activity.

Online Resources

Pass out a copy of Hello Ruby Map Algorithms to each student. Project the map on the board, and describe the task that Ruby is trying to accomplish. She is trying to visit each of her four friends on separate trips. She has to avoid the blue water by crossing at the brown bridge only.

Device-Based Activity

When everyone has completed their four algorithms, have students pair up to compare their solutions. Invite them to determine if there was only one way to navigate Ruby to each of her friends. If they had different solutions, which used the least number of steps?

## 3. Check for understanding.

Unplugged Activity

Ask students what would happen if each step in Ruby's path were not specific and accurate. Would she be able to reach each of her friends? Would she fall in the water?

**CHALLENGE**—Challenge students to work in pairs to each write one long algorithm to take Ruby to visit each of her four friends, one after the other. Pairs then compare their algorithm with others and discuss which route was the most efficient.

**EXTEND**—Extend learning by having the class direct a classmate who is role-playing a robot around the classroom using only hand signals from the rest of the class to get them from one point in the room to another. A new starting and ending point can be used, and another student becomes the robot.

## Know the Standards

*1A-AP-08 Model daily processes by creating and following algorithms (sets of step-by-step instructions) to complete tasks.* and *1A-AP-11 Decompose (break down) the steps needed to solve a problem into a precise sequence of instructions.* Decomposition is the act of breaking down tasks into simpler tasks. Composition is the combination of smaller tasks into more complex tasks. Isolating the discrete steps within every task is an important part of early algorithmic thinking.

LESSON 8, PART 1

# Steps Can Repeat

**Driving Question:** How do programmers build computer programs?

**Overarching Unit Objective:** Model use of symbols, codes, signals, and steps in simple processes and problem-solving.

## At a Glance

### Learning Objectives

- ✓ Describe sequences.
- ✓ Recognize that sequences can be repeated in loops.

### Instructional Activities

- student reading
- whole-class discussion

### CSTA Standards

**Concept:** Algorithms and Programming

**Subconcept and Standard:** **Control: 1B-AP-10** Create programs that include sequences, events, loops, and conditionals. (P5.2)

For detailed information about the CSTA Standards, follow the links in the Online Resources Guide for this unit:

**www.coreknowledge.org/cksci-online-resources**

## Core Vocabulary and Language of Instruction

A Glossary at the end of this Teacher Guide lists definitions for Core Vocabulary and selected Language of Instruction.

**Core Vocabulary** terms are those that students should learn to use accurately in discussion. During instruction, expose students repeatedly to these terms but not through isolated drill or memorization.

**decompose** **loop** **program** **sequence**

**Language of Instruction** consists of additional terms that you should use when talking about concepts in the lesson. Students benefit from your modeling the use of these words without the expectation that students will use or explain the words themselves.

**break down** **commands** **repeat**

## Instructional Resource

Student Reader

Ch. 8

**Student Reader, Chapter 8**

Steps Can Repeat

## Materials and Equipment

**Collect or prepare the following:**

- classroom computer, tablet, or other computer device for discussion

# The Core Lesson

## 1. Focus attention on today's topic.

Why are loops important? Explain that people create computer programs by composing sequences of commands that specify the precise order in which instructions should be executed. Programs are designed by breaking tasks into smaller parts and recombining existing solutions or using loops. Loops enable programs to repeat a sequence of commands multiple times without having to write the code over and over.

## 2. Read together "Steps Can Repeat"

Student Reader

Ch. 8

Read together or have students read independently "Steps Can Repeat" Chapter 8 in the Student Reader. The selection introduces the idea that repeating steps is a necessary process of creating program code. However, loops allow the creator to repeat a command without having to write the same code over and over.

### Guided Reading Supports

When reading aloud together as a class, always prompt students to follow along. Pause for discussion. Include suggested questions and prompts:

**Page 34**

How is baking cookies just like programming a task for Algo?

» When Jenni bakes cookies, there are steps that must be performed in a specific order.

What would happen if the steps to bake cookies were not done in the correct order?

» The cookies would not turn out looking like cookies, or they might not taste good.

**Page 35**

What task frustrates Jenni and why?

» Jenni is tired of scooping the dough from the bowl to the pan because she knows she has to do it three dozen times!

What does Jenni realize as she is scooping the dough over and over?

» Jenni realizes she has to enter the same code many times when she wants Algo to complete a task over and over.

**Page 36**

What does Jenni learn about Algo repeating a sequence of steps?

» She learns that loops allow her to program a sequence of steps just once but that she can instruct Algo to repeat it as many times as she wants.

How do loops help Jenni?

» It saves her a lot of time because she does not have to enter the same bit of code over and over.

**Page 37**

How many times will Algo perform the loop of these commands?

- crouch down
- lower his head
- make a licking sound
- then stand up

» Students count four times in the maze where Algo will perform the loop.

## 3. Check for understanding.

**Unplugged Activity**

Ask students to describe something in their lives that they repeat over and over. They might mention brushing their teeth or getting ready for school. Have them describe what it would feel like to put these activities on a loop.

**CHALLENGE**—Challenge students to create their own maze that incorporates a repeated sequence of steps at least three times.

**EXTEND**—Extend student learning by prompting students to program each other like a robot by giving a sequence of three steps. They can then give other instructions, but their robot classmates must perform the sequence if LOOP is called.

## Know the Science

A loop is a control structure that repeats a sequence of instructions multiple times. In programming, many times loops will be set to occur as long as a specific condition is true. For example, a question might be asked, and any time the answer is "true," the computer might be instructed to complete a loop of steps. If the answer to the question is "false," the loop would not take place.

LESSON 8, PART 2

# Sequences and Loops

**Driving Question:** How do programmers build computer programs?

**Overarching Unit Objective:** Model use of symbols, codes, signals, and steps in simple processes and problem-solving.

## At a Glance

### Learning Objective

✓ Create a program with a sequence of commands and simple loops of repeating steps.

### Instructional Activities

- whole-class discussion
- class activity

### CSTA Standards

**Concept:** Algorithms and Programming

**Subconcept and Standard:** **Control: 1B-AP-10** Create programs that include sequences, events, loops, and conditionals. (P5.2)

For detailed information about the CSTA Standards, follow the links in the Online Resources Guide for this unit:

**www.coreknowledge.org/cksci-online-resources**

## Core Vocabulary and Language of Instruction

A Glossary at the end of this Teacher Guide lists definitions for Core Vocabulary and selected Language of Instruction.

**Core Vocabulary** terms are those that students should learn to use accurately in discussion. During instruction, expose students repeatedly to these terms but not through isolated drill or memorization.

**decompose** **loop** **program** **sequence**

**Language of Instruction** consists of additional terms that you should use when talking about concepts in the lesson. Students benefit from your modeling the use of these words without the expectation that students will use or explain the words themselves.

**break down** **commands** **repeat**

## Materials and Equipment

**Collect or prepare the following:**

- None

## Advance Preparation

- Make sure that there is adequate room for students to move around and do the physical tasks during the Looper Says game.

# The Core Lesson

## 1. Focus attention on today's topic.

What can you do with repeating steps? Direct students' attention to the concept of repetitive tasks in their lives, such as making their bed every morning. Invite them to list the discrete steps needed to complete this task. Ask them if they think it is a good idea to be able to say, "Make bed," and have the multiple-step task happen automatically.

## 2. Facilitate the activity.

In this kinesthetic activity, Looper Says, you begin by being the first Looper. Students stand up in place and are given instructions to do certain moves when asked to.

Unplugged Activity

As the Looper, you need to teach your class a series of three or four moves that will combine to create the LOOP. For example, "Touch your head, then touch your shoulders, then touch your knees, and then touch your toes."

The Looper gives commands such as "Looper says stand on one foot" or "Touch your nose." Just as in the traditional game, students can only move when first prompted by "Looper says."

When "Looper says LOOP" is called, students should continue the series of actions over and over again until "Looper says stop" is called. If a student moves out of turn, they must sit down in their chair or on the floor until the next round begins.

Add on to the complexity by creating multiple loops for students to follow. For example, LOOP A is as listed above, but LOOP B is hand on right hip, then hand on left hip, then hip circle.

Students can take turns creating LOOP sequences and leading the whole class or smaller groups as the Looper.

## 3. Check for understanding.

Unplugged Activity

Ask students to each create a series of three simple moves of their own and consider this their own personal LOOP. You once again become the Looper. And this time when "Looper says LOOP" is called, each student should be repeating their very own sequence of moves.

**CHALLENGE**—Challenge students to find sequences and loops in their extracurricular activities, such as art, music, or sports. For example, when they play a song on the piano, is there a repeating chorus? If they are sewing a craft, is there a repeating move they do with the needle? Do they have drills they repeat during soccer or basketball practice?

**EXTEND**—Extend the activity to the playground by creating longer and larger motor sequences of moves for students. For example, a hop on one foot, three large steps, a skip, and then one jumping jack.

## Know the Standards

*1A-AP-10 Develop programs with sequences and simple loops, to express ideas or address a problem.* As students learn to develop computer programs with simple step-by-step sequences, or algorithms, they will soon realize that repetitive tasks are common in programming. Loops are essential to saving time and minimizing errors. Programmers can shorten hundreds of lines of code by using loops, which can be used as many times as needed.

LESSON 9, PART 1

# Series of Commands

**Driving Question:** How do programmers build computer programs?

**Overarching Unit Objective:** Model use of symbols, codes, signals, and steps in simple processes and problem-solving.

## At a Glance

### Learning Objective

✓ Identify steps computer programmers use in the design process.

### Instructional Activities

- student reading
- whole-class discussion

### CSTA Standards

**Concept:** Algorithms and Programming

**Subconcept and Standards:**

**Program Development: 1B-AP-13** Use an iterative process to plan the development of a program by including others' perspectives and considering user preferences. (P1.1, P5.1)

**Program Development: 1B-AP-14** Observe intellectual property rights and give appropriate attribution when creating or remixing programs. (P7.3)

**Program Development: 1B-AP-15** Test and debug (identify and fix errors) a program or algorithm to ensure it runs as intended. (P6.1, P6.2)

**Program Development: 1B-AP-16** Take on varying roles, with teacher guidance, when collaborating with peers during the design, implementation, and review stages of program development. (P2.2)

**Program Development: 1B-AP-17** Describe choices made during program development using code comments, presentations, and demonstrations. (P7.2)

For detailed information about the CSTA Standards, follow the links in the Online Resources Guide for this unit:

**www.coreknowledge.org/cksci-online-resources**

## Core Vocabulary and Language of Instruction

A Glossary at the end of this Teacher Guide lists definitions for Core Vocabulary and selected Language of Instruction.

**Core Vocabulary** terms are those that students should learn to use accurately in discussion. During instruction, expose students repeatedly to these terms but not through isolated drill or memorization.

**credit** **design** **programmer**

**Language of Instruction** consists of additional terms that you should use when talking about concepts in the lesson. Students benefit from your modeling the use of these words without the expectation that students will use or explain the words themselves.

**iterative** **testing**

## Instructional Resource

Student Reader

Ch. 9

**Student Reader, Chapter 9**

Series of Commands

## Materials and Equipment

**Collect or prepare the following:**

- classroom computer, tablet, or other computer device for discussion

# The Core Lesson

## 1. Focus attention on today's topic.

What is a computer programmer? Explain that computer programmers are people who develop programs collaboratively and for a purpose, such as expressing ideas or addressing problems. Explain that people work together to plan, create, and test programs to ensure they work as intended. Highlight the steps of the design process: Ask, Imagine, Plan, Create, Test, Improve, Share.

## 2. Read together "Series of Commands"

Student Reader

Ch. 9

Read together or have students read independently "Series of Commands," Chapter 9 in the Student Reader. The selection introduces the idea that working on the design process is a collaborative activity.

### Guided Reading Supports

When reading aloud together as a class, always prompt students to follow along. Pause for discussion. Include suggested questions and prompts:

**Page 38**

Why do you think Jenni likes solving problems with a friend?

» Answers will vary. Possible answer: She feels that working together gives her new ideas to try.

**Page 39**

What is Nataly's idea for Algo?

» Nataly wants to design a cat yoga routine for Algo.

What are the steps the friends take to make Nataly's idea happen?

» They work together to plan each move Algo will make and in which order, and then they code those steps into his controller in the proper sequence. They use loops wherever possible.

**Page 40**

Why do Jenni and Nataly try out Algo's routine many times?

» They are adjusting their code to be sure it works just the way they want it to.

Why does Jenni's mom try out Algo's routine?

» Jenni's mom helps them test Algo's routine to be sure it is a successful yoga stretch.

**Page 41**

Why does Jenni credit Nataly?

» Jenni knows it is important to give credit to anyone who helps think of and create a program, such as Algo's yoga routine.

What steps did the girls go through to complete their design?

» They first imagined creating a cat yoga routine, then they planned out the moves, coded the moves, tested the routine many times, adjusted and improved their design, and finally shared their program with Jenni's mom.

## 3. Check for understanding.

Unplugged Activity

Ask students to raise their hands and define what a computer programmer does and determine whether they work alone. Ask them to name and describe one of the steps in the design process.

**CHALLENGE**—Challenge students by asking them to break down each step of a move they do in an extracurricular activity, such as music or sports. Each student demonstrates the task to the class, narrating each step with "Step 1," "Step 2," and so on.

**EXTEND**—Extend student learning by having students work in teams to design and code a sequence of moves on paper and then test out the routines with other teams. Teams should provide each other positive feedback and suggestions for improvement.

## Know the Science

Online Resources

While there are many variations of the steps in the design process, understanding the basic components of design thinking is critical in computer science. Researching or imagining a solution, defining the goals, creating a plan, coding or building a solution, testing and iterating the protype, and sharing with others are just a few terms used for this important process used to design solutions to any type of problem. See the Online Resource Guide for a link to learn more about the design thinking process.

**www.coreknowledge.org/cksci-online-resources**

LESSON 9, PART 2

# Fun Programming

**Driving Question:** How do programmers build computer programs?

**Overarching Unit Objective:** Model use of symbols, codes, signals, and steps in simple processes and problem-solving.

## At a Glance

### Learning Objective

✓ Use an online application to compose a sequence of commands to program actions of animated characters.

### Instructional Activities

- supervised individual computer operation
- group reflection

### CSTA Standards

**Concept:** Algorithms and Programming

**Subconcepts and Standards:**

**Control: 1B-AP-10** Create programs that include sequences, events, loops, and conditionals. (P5.2)

**Modularity: 1B-AP-12** Modify, remix, or incorporate portions of an existing program into one's own work, to develop something new or add more advanced features. (P5.3)

For detailed information about the CSTA Standards, follow the links in the Online Resources Guide for this unit:

**www.coreknowledge.org/cksci-online-resources**

## Core Vocabulary and Language of Instruction

A Glossary at the end of this Teacher Guide lists definitions for Core Vocabulary and selected Language of Instruction.

**Core Vocabulary** terms are those that students should learn to use accurately in discussion. During instruction, expose students repeatedly to these terms but not through isolated drill or memorization.

**commands** **programming** **sequence**

**Language of Instruction** consists of additional terms that you should use when talking about concepts in the lesson. Students benefit from your modeling the use of these words without the expectation that students will use or explain the words themselves.

**sprite**

## Materials and Equipment

**Collect or prepare the following:**

- computer and internet access to the recommended programming app (ScratchJr) for each student

(See the online Resources Guide for links to these items. **www.coreknowledge.org/cksci-online-resources**)

- means to project the teacher's screen for demonstration and student guidance

## Advance Preparation

- Practice using the coding application in advance of introducing students to it. ScratchJr is an introductory programming language that enables young children to create their own interactive stories and games. ScratchJr is available as a free app. For full information, including tutorials, please visit https://www.scratchjr.org/about/info.
- Use your school's protocol for logging in and making sites available for students.
- Set aside multiple class sessions for students to practice with the application. The more interactive sessions you can provide students with, the greater their skills will be reinforced.

# The Core Lesson

## 1. Focus attention on today's topic.

Online Resources

Tell students that in today's actvity they will program animated characters to follow computer commands. They will use the commands to create a dance routine for characters in the program. First, tell students you will program them to dance! Have students to stand up in an open area or by their desks or tables. Hold the motion blocks up one-by-one as you walk the students through the dance program. Have students practice the movements as you point to each motion block. Display the entire program using the motion blocks and have students do the dance. Finally, project your computer screen for students to view and show the dance party you programmed in ScratchJr. Help students connect to the idea that when programming, they will have to plan out each step and these will all come together to become the final product.

## 2. Facilitate the coding activity.

Device-Based Activity

Project the "Quick Intro to ScratchJr" video found on the front page of the app. It is marked with a "?" on the front page of the app. Watch the video as a class. If desired, provide students with physical copies of the Interface Guide for reference. The Blocks Guide is also a helpful resource within the app that lists and describes the different blocks that can be used. Tell students that in this activity, they will be working as programmers, and this guide lists the different topics they can choose from.

Have students, either individually or in pairs, log on to ScratchJr. Project the directions for the dance party from https://connect.scratchjr.org/classroom-dance-party/. Proceed through the instructions as a class, stopping occasionally to discuss options and to troubleshoot.

When dance routines are complete have students share their final dance parties with a peer. As they watch the routines, they should compare the choices they made and discuss possible improvements.

**SUPPORT**—Before creating the dance party, it may be helpful to allow students to explore the ScratchJr. app. Set aside five to ten minutes to allow students to navigate and play with the app.

**SUPPORT**—If students are having trouble choosing blocks, have them first build their programs using their own physical copies of the motion blocks. Students can choose the blocks they think they will use, set them out in sequential order, and then physically act them out to decide if these are the movements they want to program. Once they have settled on the sequence, they can program the motions into the app.

## 3. Check for understanding.

Online Resources

Unplugged Activity

Convene the class for discussion after students have had ample time for programming experience. You may choose to display some student outcomes for discussion. Prompt discussion with students using questions like these:

- What were you able to do using the programming application?
- What did you enjoy most about it?
- Was any of it hard to do?
- What would make that easier?
- What would you like to do next?

**EXTEND**—If additional time is available, have students choose a project to create from the ScratchJr Activities database found on https://www.scratchjr.org/teach/activities. Once they have learned the new techniques introduced in these activities, have them revise or update their dance party routine using thenew technique.

**EXTEND**—Have students design and develop their own program. First, students should determine the goal of their program. For example, they can design and develop a car race, a beach scene, or create a short story.

LESSON 9, PART 3

# Solving Design Problems

**Driving Question:** How do programmers build computer programs?

**Overarching Unit Objective:** Model use of symbols, codes, signals, and steps in simple processes and problem-solving.

## At a Glance

### Learning Objectives

- ✓ Identify a problem or task and break it into incremental steps.
- ✓ Apply the steps of the design process to solve a problem.

### Instructional Activities

- class discussion
- small-group activity

### CSTA Standards

**Concept:** Algorithms and Programming

**Subconcept and Standard:** **Program Development: 1B-AP-16** Take on varying roles, with teacher guidance, when collaborating with peers during the design, implementation, and review stages of program development. (P2.2)

For detailed information about the CSTA Standards, follow the links in the Online Resources Guide for this unit:

**www.coreknowledge.org/cksci-online-resources**

## Core Vocabulary and Language of Instruction

A Glossary at the end of this Teacher Guide lists definitions for Core Vocabulary and selected Language of Instruction.

**Core Vocabulary** terms are those that students should learn to use accurately in discussion. During instruction, expose students repeatedly to these terms but not through isolated drill or memorization.

**credit** **debug** **design** **programmer**

**Language of Instruction** consists of additional terms that you should use when talking about concepts in the lesson. Students benefit from your modeling the use of these words without the expectation that students will use or explain the words themselves.

**iterative** **testing**

## Materials and Equipment

**Collect or prepare the following:**

- large whiteboard
- blank paper

## Advance Preparation

- None

# The Core Lesson

## 1. Focus attention on today's topic.

How can you solve a design problem? On the whiteboard, write the steps in the design process in order. Ask students to provide their definitions or point out synonyms for each of the steps: Ask, Imagine, Plan, Create, Test, Improve, Share.

## 2. Facilitate the activity.

Just like Jenni and Nataly, students get to engage in the design process to create their own robot pet action. In groups of three or four students, students ideate a physical task for their robot pet to perform that is characteristic of the type of animal they have selected. Indicate to groups that they should do the following:

Unplugged Activity

- **Imagine** the type of robot animal they would like to have (e.g., robot dog.)
- Brainstorm and **plan** what routine or behavior it will be performing (e.g., "Our robot dog will chase and fetch a ball.").
- Writing or drawing on paper, **create** at least ten discrete steps the pet will use to execute the task (e.g., "1. Pick up ball. 2. Bring ball to me. 3. Drop ball in front of me. 4. Bark. 5. Watch ball as it is thrown. 6. Chase after ball. 7. Stop ball with nose. 8. Pick up ball in mouth. 9. LOOP.").
- Students take turns in their group **testing** their program by performing each of the steps in the task and adjusting to **improve** the program as needed.

## 3. Check for understanding.

Unplugged Activity

When ready, groups should **share** by demonstrating their programs and give each other positive, constructive feedback. Students should be sure to give credit to those in their group for their contributions.

**CHALLENGE**—Challenge students to plan and create another sequence of steps for their robot pet that are not characteristic for their animal (e.g., a dog swinging from a tree or a rabbit fetching a ball).

**EXTEND**—Extend student learning by prompting them to think about how they could use the design process to find a solution to a problem at school or in the world.

## Know the Standards

*1A-AP-12 Develop plans that describe a program's sequence of events, goals, and expected outcomes.* and *1A-AP-13 Give attribution when using the ideas and creations of others while developing programs.* Developing an effective computer program is not just about the coding, but it also requires thinking through the entire design process. Introducing young children to first identify goals and expected outcomes and then plan a sequence of events prior to coding is important. Learning the importance of testing their solution and iterating until it works as expected is also a critical step to learn. In addition, sharing their design and crediting those involved in the design process help them build essential collaboration skills.

GRADE 3

# Teacher Resources

**Activity Pages**

Name _____ Date _____

Activity Page 1.2 Use with Lesson 1.2

# Computer Applications

| | |
|---|---|
| Word Processor |  |
| Spreadsheet |  |
| Presentation |  |
| Graphics |  |
| Storage |  |
| Browser |  |
| Mobile Tech |  |
| Output |  |
| Processing |  |
| Input |  |
| Messaging |  |
| Social Media |  |

# Appendix A

## Glossary

**Colored words and phrases** are Core Vocabulary for the unit. **Bold-faced words and phrases** are additional vocabulary terms related to the unit that you should model for students during instruction. Vocabulary words are not intended for use in isolated drill or memorization.

### A

**algorithm, n**. a set of steps to follow

**attribute, n**. to give credit, explain the cause of something

### B

**binary, adj**. something made up of two things

**binary code, n**. computer data made up of bits

**binary data, n**. information made from only two symbols or values

**bit, n**. a single piece of computer data; either a 1 or a 0

**break down, v**. separate into smaller parts or steps

**byte, n**. a unit of information; usually eight bits

### C

**characteristic, n**. a quality or feature of something that can be used to describe it

**code, n**. a set of symbols for communicating a message

**collect, v**. bring togther things in a group

**commands, n**. instructions

**communicate, v**. to share thoughts, ideas, or information

**component, n**. a part or piece of something

**computer, n**. an electronic device that is used to store, organize, and work with data at a very high speed

**computer code, n**. strings of symbols that make up instructions for computers to follow

**computing device, n**. any type of machine that stores, organizes, and works with digital data at a high speed, such as computers, tablets, and calculators

**composition, n**. the combination of smaller tasks into more complex tasks

**connection, n**. a link enabling data transfer between computing devices

**credit, n**. acknowledgment or recognition

**cyberbullying, v**. a person who writes hurtful messages on the internet with the goal of being mean to another person

### D

**data, n**. informatoin in digital format

**decompose, v**. break down tasks into simpler tasks

**decomposition, n**. the act of breaking down tasks into simpler tasks

**digital data, n**. information in digital format stored or processed in an electronic device

**debug, v**. to fix problems or errors

**design, v**. to plan how to arrange or construct a solution

**digital, adj**. describing information within electronic devices that is composed of binary data

**digital citizenship, n**. the practice of behaving kindly, respectfully, and safely during internet activity

**digital footprint, n**. data about an internet user that remains on other computers connected to the internet even when the user is not online

### F

**function, n**. job; what an object is meant to do

### G

**graph, n**. a diagram that organizes and displays data in a way that reveals patterns and makes the data easier to understand

### H

**hardware, n**. parts of a computer device that can be touched

### I

**icon, n**. a small image or symbol that represents an app or program

**interaction, n**. an event in which two or more things affect each other

**internet, n**. many computers all over the world that are connected together and share information with each other

## L

**language, n**. spoken or written words that are used to communicate with others

**log off, v**. to exit a website and close a user's access to a program

**loop, n**. repeated instructions

## N

**network, n**. a system of computers that communicate with each other

## O

**online, adj**. connected to the internet

## P

**password, n**. a secret sequence of characters needed to access a computer device or program

**pattern, n**. characteristics that reoccur in a similar way

**permanent, adj**. lasting or meant to last for a very long time

**personal identifiable information, n**. a number that is used to make sure only certain people can get, use, or change information

**predict, v**. to say what will happen before it happens

**prediction, n**. a statement that something might happen in the future

**privacy, n**. The ability of a person to hide information about themselves

**processing, v**. to work with data by following a procedure

**program, n**. steps or list of instructions that tells computer how to do a task

**programmer, n**. people who develop programs for a purpose

**programming, v**. developing steps or lists of instructions that tell a computer program what to do

**protection, n**. a person or thing that keeps something safe from harm

## R

**reboot, v**. to shut down and restart

**repeat, v**. to do or say the same thing over again

**represent, v**. to stand for something

**routine, n**. a regular series of steps in an activity

## S

**safety, n**. the state of being protected from harm

**screentime, n**. the amount of time spent using digital devices

**security, n**. conditions that provide protection

**sequence, n**. the order of steps or instructions

**software, n**. instructions that computer hardware follows

**step-by-step, adj**. describing the breakdown of the parts of a larger process

**storage, n**. the capacity for saving something for use later

**symbol, n**. a mark that mean something

## T

**task, n**. an action that needs to be completed

**technology, n**. the use of scientific knowledge in devices or processes

**testing, n**. the process of trying out a solution in a controlled way to see if it works as planned

## U

**unauthorized, adj**. without permission

**update, n**. new instructions to change or improve performance of a computing device

**username, n**. a unique sequence of characters that identifies a user of a computer program

## V

**visualize, v**. to form a picture of something in the mind

**visualization, n**. an image that formed to help an idea or a set of data be more understandable

# Appendix B

## Internet Safety

Though online resources present many rich opportunities for student learning, unsupervised online activity for children is not advised. The U.S. Department of Justice provides the following guidelines, Keeping Children Safe Online:

- **Discuss internet safety and develop an online safety plan** with children before they engage in online activity. Establish clear guidelines, teach children to spot red flags, and encourage children to have open communication with you.
- **Supervise young children's use of the internet**, including periodically checking their profiles and posts. Keep electronic devices in open, common areas of the home, and consider setting time limits for their use.
- **Review games, apps, and social media sites** before they are downloaded or used by children. Pay particular attention to apps and sites that feature end-to-end encryption, direct messaging, video chats, file uploads, and user anonymity, which are frequently relied upon by online child predators.
- **Adjust privacy settings and use parental controls** for online games, apps, social medial sites, and electronic devices.
- **Tell children to avoid sharing personal information, photos, and videos online** in public forums or with people they do not know in real life. Explain to your children that images posted online will be permanently on the internet.
- **Teach children about body safety and boundaries**, including the importance of saying "no" to inappropriate requests both in the physical world and the virtual world.
- **Be alert to potential signs of abuse**, including changes in children's use of electronic devices, attempts to conceal online activity, withdrawn behavior, angry outbursts, anxiety, and depression.
- **Encourage children to tell a parent, guardian, or other trusted adult** if anyone asks them to engage in sexual activity or other inappropriate behavior.

Copy and distribute the Student Safety Contract, found on the next page. Prior to the start of the first lesson, do a read-along, and have students agree to the expectations for when they engage in computer and online activities.

Online Resources

For additional support for safety in the computer science and online instruction, follow the links in the Online Resources Guide for this unit:

**www.coreknowledge.org/cksci-online-resources**

# Student Online Safety Contract

Dear Parent or Guardian,

During computer science class, we want to create and maintain a safe classroom. With this in mind, we are making sure students are aware of the expectations for their behavior while engaged in computer science activities. We are asking you to review the safety rules below with your student and sign this contract. If you have any questions, please feel free to contact me.

For important safety information about children, computers, and the internet, consider resources at these sites:

https://protectyoungeyes.com/

https://sharedhope.org/

https://www.justice.gov/coronavirus/keeping-children-safe-online

_____ _____/_____/_____/

Teacher signature and date

_____ _____/_____/_____/

Parent or guardian signature and date

---

When doing online activities, I will do the following:

- Only do online activities with the supervision of an adult.
- Only visit websites and use apps that I am guided to by my teacher, parent, or trusted adult guardian.
- Never use my real name or reveal personal information if I communicate with others online.
- Tell a trusted adult right away if anyone online asks questions about my name, where I live, or where I go to school.
- Be careful around electronic devices and only plug them in or unplug them when an adult is supervising.

I understand and agree to the safety rules in this contract.

_____ _____/_____/_____/

Student signature and date

_____

Print name

# Appendix C

## Strategies for Acquiring Materials

The materials used in the Core Knowledge Computer Science program are readily available and can be acquired through both retail and online stores. Some of the materials will be reusable and are meant to be used repeatedly. This includes items such as plastic cups that can be safely used again. Often these materials are durable and will last for more than one activity or even one school year. Other materials are classified as consumable and are not able to be used more than once.

Online Resources

The Material Supply List for this unit's activities can be found online. Follow the links in the Online Resources Guide for this unit:

**www.coreknowledge.org/cksci-online-resources**

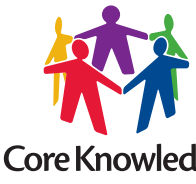
### Ways to Engage with Your Community

The total cost of materials and technology can add up for an entire unit, even when the materials required for activities and demonstrations have been selected to be individually affordable. And the time needed to acquire the materials adds up too. Reaching out to your community to help support STEM education is a great way to engage parents, guardians, and others with the teaching of science and computer science, as well as to reduce the cost and time of collecting the materials. With that in mind, the materials list can be distributed or used as a reference for the materials teachers will need to acquire to teach the unit.

**Consider some of the following as methods for acquiring the science materials:**

- School Supply Drive—If your school has a supply drive at any point in the year, consider distributing materials lists as wish lists for the science department.
- Open Houses—Have materials lists available during open houses. Consider having teams of volunteers perform an activity to show attendees how the materials will be used throughout the year.
- Parent-Teacher Organizations—Reach out to the local PTO for assistance with acquiring materials.
- Science Fair Drive—Consider adding a table to your science fair as part of a science materials drive for future units.
- College or University Service Project—Ask service organizations affiliated with your local higher education institutions to sponsor your program by providing materials.
- Local Businesses—Some businesses have discounts for teachers to purchase school supplies. Others may want to advertise as sponsors for your school/programs. Usually you will be asked for verifiable proof that you are a teacher and/or for examples of how their sponsorship will benefit students.

Remember: If your school is public, it will be tax exempt, so make sure to have a Tax Identification Number (TIN) when purchasing materials. If your school is private, you may need proof of 501(c)(3) status to gain tax exemption. Check with your school for any required documentation.




CKSci™

Core Knowledge SCIENCE™

**Editorial Director**

Daniel H. Franck

## Subject Matter Expert

Sarah J. Huibregtse, PhD

## Illustrations and Photo Credits

Michele D'Ottavio / Alamy Stock Photo: Cover B

Veronika Pfeiffer / Alamy Stock Photo: Cover A

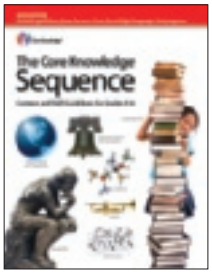
Core Knowledge Curriculum Series™

CKSci™

Core Knowledge SCIENCE™

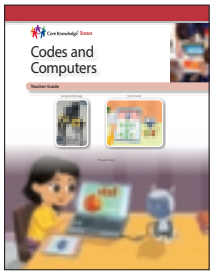
# Codes and Computers

## Core Knowledge Science 3




**What is the Core Knowledge Sequence?**

The *Core Knowledge Sequence* is a detailed guide to specific content and skills to be taught in Grades K–8 in language arts, history, geography, mathematics, science, computer science, and the fine arts. In the domain of computer science, the *Core Knowledge Sequence* outlines topics that build systematically grade by grade to support student learning progressions coherently over time.




**For which grade levels is this book intended?**

In general, the content and presentation are appropriate for students in the upper elementary grades. For teachers and schools following the *Core Knowledge Sequence*, this book is intended for Grade 3 and is part of a series of **Core Knowledge SCIENCE** units of study.

For a complete listing of resources in the **Core Knowledge SCIENCE** series, visit **www.coreknowledge.org**.

CKSci™

Core Knowledge SCIENCE™

A comprehensive program in science, integrating topics from
Earth and Space, Life, and Physical Sciences
with concepts specified in the Core Knowledge Sequence
(content and skill guidelines for Grades K–8).

Core Knowledge SCIENCE™
units at this level include:

**Investigating Forces**
**Life Cycles, Traits, and Variations**
**Habitats and Change**
**Weather and Climate**
**Human Senses and Movement**
**Codes and Computers**

**www.coreknowledge.org**

Core Knowledge Curriculum Series™

ISBN: 978-1-68380-992-0