

Course B

Course B was developed with first graders in mind. Tailored to a novice reading level, this course also assumes limited knowledge of shapes and numbers.

At the moment, Course B closely parallels Course A, but provides more complex unplugged activities and more variety in puzzles. Students will learn the basics of programming, collaboration techniques, investigation and critical thinking skills, persistence in the face of difficulty, and internet safety. At the end of this course students will create their very own custom game from Play Lab that they can share with a link.

Teacher Links: [Teacher Videos](#) [Playlist](#)

Week 1

Lesson 1: Move It, Move It

Lesson 2: Sequencing with Scrat

Lesson 3: Your Digital Footprint

Common Sense Education | Unplugged

Lesson 4: My Robotic Friends Jr.

Algorithms | Debugging | Unplugged

Lesson 5: Programming with Scrat

Algorithms | Debugging | Program | Programming

Lesson 6: It's Great to Create and Play Fair

Lesson 7: Programming with Rey and BB-8

Programming | Maze

Lesson 8: My Loopy Robotic Friends Jr.

Unplugged | Loop | Repeat

Lesson 9: Loops with Scrat

Loop | Ice Age | Scrat

Lesson 10: Loops with Laurel

Loop | Collector

Lesson 11: Drawing Gardens with Loops

Loop | Artist

Lesson 12: The Big Event Jr.

Event | Unplugged

Lesson 13: A Royal Battle with Events

Event | Play Lab



This curriculum is available under a
Creative Commons License (CC BY-NC-SA 4.0).

If you are interested in licensing Code.org materials for commercial purposes, **contact us**.

Lesson 1: Move It, Move It

Overview

This lesson will work to prepare students mentally for the coding exercises that they will encounter over the length of this course. In small teams, students will use physical activity to program their classmates to step carefully from place to place until a goal is achieved.

Purpose

By using physical movement to program their classmates, students will run into issues and emotions similar to what they will feel when they begin coding on a computer. Encountering those stresses in a playful and open environment will help to alleviate intensity and allow students to practice necessary skills before they run into problems on their own.

Agenda

Warm Up (20 min)

Where did I go wrong?

Activity: Move It, Move It (20 minutes)

Wrap-up (10 min)

Journaling

Extended Learning

View on Code Studio

Objectives

Students will be able to:

- Define a list of steps (algorithm) to get a friend from their starting position to their goal
- Translate a list of steps into a series of physical actions
- Identify and fix errors in the execution of an algorithm

Preparation

- Watch the **Teacher Video - Move it, Move it**
- Print (or otherwise prepare) one **Move It, Move It Teacher Debugging Puzzle - PDF** for displaying to the class
- Print one **Move It, Move It: Debugging - PDF** for each group of 2-3 students
- Print one **Move It, Move It: Activity - pdf** for each group of 2-3 students
- Prepare blank papers to fill out the rest of the walking grid (4-7 needed per group)

Links

Heads Up! Please make a copy of any documents you plan to share with students.

For the Teacher

- **Move It, Move It Teacher Debugging Puzzle - PDF**
- **Teacher Video - Move it, Move it**

For the Students

- **Move It, Move It: Activity - pdf**
- **Move It, Move It: Debugging - PDF**

Vocabulary

- **Algorithm** - A precise sequence of instructions for processes that can be executed by a computer
- **Bug** - Part of a program that does not work

correctly.

- **Debugging** - Finding and fixing problems in an algorithm or program.
- **Frustrated** - Feeling annoyed or angry because something is not the way you want it.
- **Persistence** - Trying again and again, even when something is very hard.

Teaching Guide

Warm Up (20 min)

Where did I go wrong?

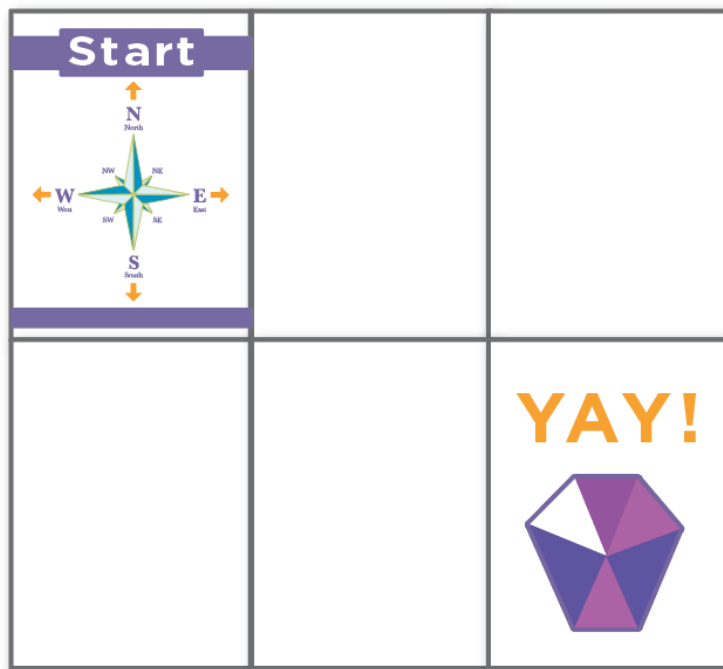
📍 **Goal:** In this lesson, we want to help students learn to identify and fix bugs in their own programs. The easiest way to do that is to first present students with a program that contains bugs that are not their fault. Once they've helped you fix "your" program, share with them how frustrating it can be to make mistakes, and help them see that those feelings are completely normal and they shouldn't feel embarrassed by them.

Display: Put an image of **Move It, Move It Teacher Debugging Puzzle - PDF** up on the screen where everyone can see it.

💡 Teaching Tip

If your class has not already learned cardinal directions, it will be worth covering them before they begin Course B. This conversion will come in handy for nearly all of the online puzzles aimed at first grade, as well as several of the unplugged activities.

Let students know that they will continue to see those directions in the online programs next to the direction arrows.



Move East →

Move East →

Move North ↑

Discuss: Get the attention of the class and let them know that you are stuck! You have this challenge, and you thought you had solved it, but it doesn't seem to be working. Your program has a bug, can they help you fix it?

Take a moment to walk them through the rules: - Start at the compass rose - Follow the instructions step-by-step - End at the happy face

Optional: Walk through your program using your fingers on top of the map, under the document camera. Express frustration when your fingers end up off the map, instead of at the treasure.

Think: My program says "East, East, North". Can you figure out why my program doesn't work?

Pair: Let students work together to see if they can figure out what the program is supposed to say.

Share: Ask students if anyone was able to figure out a way to solve the problem. When you get a correct answer, let the students know that they are great at "debugging"!

Discuss: Ask the students if they could tell how you were feeling when you couldn't figure out the answer. They might suggest that you were "mad" or "sad". Instead of telling them "no", describe that you were feeling a little bit mad, a little bit sad, and a little bit confused. When you put all of those emotions together, it makes a feeling called "frustration". When you are "frustrated" you might think you are mad, sad, or confused -- and you might be tempted to give up -- but frustration is a natural feeling and it's a big hint that you are about to learn something! Instead of quitting, practice persistence. Keep trying over and over again. After a few times, you will start to understand how to debug your problems!

Distribute: To make sure that students understand the idea of finding and fixing errors (debugging) pass out the **Move It, Move It: Debugging - PDF** and have students complete the task in pairs.

Optional: If you want to move the activity along more quickly, feel free to complete these as a class, instead.

Transition: Now it's time to play the game!

Content Corner

For more on persistence and frustration, try reading **Stevie and the Big Project** to your students. It will help them spot moments of frustration. It will also help give them the tools to deal with it.

If you do not read the book, take a moment to cover tips on frustration and persistence as a class:

Tips to Help With Frustration

- Count to 10
- Take deep breaths
- Journal about them
- Talk to a partner about them
- Ask for help

Tips for Being Persistent

- Keep track of what you have already tried
- Describe what is happening
- Describe what is supposed to happen
- What does that tell you?
- Make a change and try again

Activity: Move It, Move It (20 minutes)

Distribute: Hand each group of 2-3 students a packet of Move It, Move It maps, as well as the blank papers for the grid on the ground. Allow students to either cut the halves of each map apart, or fold the sheets in half so that each map is clearly visible (without distraction.)

Set-Up: In each group, each player will get a task.

- Player 1: Choose/set-up the map to play
- Player 2: Programmer
- Player 3: Walking Machine

Directions for Class:

- 1) Decide who will take each job.
- 2) Have player 1 set a grid on the floor made up of pieces of paper (as shown on one of the Move It Maps) except with the gem paper facing the ground.
- 3) Player 3 will start by standing on the page with the compass rose.
- 4) Player 2 will then guide player 3 step-by-step through the paper maze using the provided arm signals.
- 5) When player 2 gives the signal to "STOP", player 3 will flip over the page that they are on. If that page is a gem, then the maze was a success!
- 6) If there is time, let everyone rotate positions and go again!

Note that the rules are not the most important thing here. Feel free to clarify if the students have questions, but if the students are playing a bit differently than described, you don't need to hold them to the letter of the game. The crucial bit is that they are moving from immediate instructions to giving two or three instructions before the Walking Machine moves.

Wrap-up (10 min)

Journaling

Give the students a journal prompt to help them process some of the things that they encountered during the day. You can choose one of the prompts below, or make up

Teaching Tip

Here are some useful links in case your class hasn't yet talked about the compass rose and cardinal directions:

- **The Cardinal Directions Geography Song**
- **Cardinal Direction Mnemonics Lesson**

your own.

Journal Prompts:

- Draw a feeling face in the corner of your journal page
- What were the four directions on the compass rose?
 - What tricks can we use to remember North, South, East and West?
- Draw another way we could we have given instructions without using our arms
- Draw your favorite part about that game

Extended Learning

Use these activities to enhance student learning. They can be used as outside of class activities or other enrichment.

X's and O's

- Draw a tic-tac-toe board for the class.
- Place a single X and a single O somewhere on the board.
- Ask the class if they can get the X to the O using arm gestures as a class.

X's, O's, and Arrows

- Similar to the activity above, but have the students write their programs in advance using arrows instead of hand gestures.
 - This can be done in groups.
 - Groups can share their solutions for the class.



This curriculum is available under a
Creative Commons License (CC BY-NC-SA 4.0).

If you are interested in licensing Code.org materials for commercial purposes, **contact us**.

Lesson 2: Sequencing with Scrat

Overview

This lesson will give students an idea of what to expect when they head to the computer lab. It begins with a brief discussion introducing them to computer lab manners, then they will progress into using a computer to complete online puzzles.

Purpose

The main goal of this lesson is to build experience with computers. By covering the most basic computer functions such as clicking, dragging, and dropping, we are creating a more equal playing field in the class for future puzzles. This lesson also provides a great opportunity to introduce appropriate computer lab behavior.

Agenda

Warm Up (10 min)

Behaving in the Computer Lab

Bridging Activity - Drag and Drop (10 - 15 min)

Dragging and Dropping Algorithms Previewing Online Puzzles as a Class

Main Activity (20 - 30 min)

Course B Online Puzzles - Website

Wrap Up (5 - 10 min)

Journaling

Extension Activities

View on Code Studio

Objectives

Students will be able to:

- Model proper computer lab behaviors
- Experiment with standard block-based programming actions such as: clicking, drag and drop, etc.

Preparation

- Watch the **How to Make a Class Section on Code.org - Teacher Video**. Create a class section and make sure every student has a card with their passcode on it
- Have the school IT person add a quick link for your class section to the computer desktop
- Print out one **Move It, Move It: Activity - pdf** for use in the bridging activity
- Cut out direction blocks from **Unplugged Blockly Blocks (Grades K-1) - Manipulatives** to use with the Move It, Move It Map
- Make sure each student has a **Think Spot Journal - Reflection Journal**

Links

Heads Up! Please make a copy of any documents you plan to share with students.

For the Teacher

- **Course B Online Puzzles 2018** - Website
- **CS Fundamentals Main Activity Tips** - Lesson Recommendations

For the Students

- **Move It, Move It: Activity** - pdf
- **Feeling Faces** - Emotion Images
- **Pair Programming** - Student Video
- **Unplugged Blockly Blocks (Grades K-1)** - Manipulatives
- **Think Spot Journal** - Reflection Journal

Vocabulary

- **Click** - Press the mouse button
- **Double-Click** - Press the mouse button very quickly
- **Drag** - Click your mouse button and hold as you move the mouse pointer to a new location
- **Drop** - Release your mouse button to "let go" of an item that you are dragging

Teaching Guide

Warm Up (10 min)

Behaving in the Computer Lab

This discussion will teach students what to expect and how to behave when they enter the computer lab.

Discuss:

Have a good discussion around your computer lab expectations to make sure that students understand the rules. Some topics of discussion might include:

- Is running in the computer lab okay?
- How loudly should we walk when we are in the computer lab?
- What should you do if you get stuck on a puzzle?
- If you get frustrated, will it help to hit the computer?
- When we're about to go to the computer lab, how should we get ready?

Some possible things to cover:

- Use calm bodies in the lab
- Remember not to chew gum or candy
- Sanitize your hands
- Sit with your partner at one computer
- Make sure that the first "driver" can reach the mouse
- When you get frustrated, don't hit or shake the computer or monitor
- Follow the **20/20/20 - Website** rule
- How to deal with the **Wiggles** every 20-30 minutes (requires a free login on GoNoodle)
- Ask your partner before you ask the teacher
- Keep volume down so everyone else can hear their partners
- Use your journal for keeping track of feelings and solutions

Bridging Activity - Drag and Drop (10 - 15 min)

Choose **one** of the following to do with your class:

Dragging and Dropping Algorithms

Print out one of the Move It Maps from the "Move It, Move It" activity and display it for the students to see. On a projector or in front of the class, put some direction blocks from the **Unplugged Blockly Blocks (Grades K-1) - Manipulatives** in random order and practice "dragging and dropping" by pressing your finger on one of the paper pieces and moving it across the screen. Explain that you can "click" to select this block by tapping your finger on it, or you can "drag" the block by pressing your finger on it and moving it. To "drop" the block, release your finger.

After showing this to the class, ask for volunteers to create an algorithm for the Move It Map by "dragging and dropping" the necessary blocks.

Previewing Online Puzzles as a Class

Project a puzzle from the **Course B Online Puzzles - Website** corresponding to this lesson. Show the class how to click on a block and place it in the correct spot by dragging and dropping. Purposely make mistakes such as clicking the background or dropping the image before it's at the right spot. Ask for help from volunteers in the class when you run into these problems, and help them use the skills that they developed in the last unplugged lesson to make things right.

Main Activity (20 - 30 min)

Course B Online Puzzles - Website

This will teach students how to use Code.org to complete online puzzles.

This stage was designed to give students the opportunity to practice hand-eye coordination, clicking, and drag & drop skills. Students will also play with sequence.

The vocabulary introduced in this lesson becomes relevant during this activity. Take some time to explicitly teach how to click, double-click, drag, and drop. It might work better for you to cover these words in the classroom environment where you can lead by example -- or it might make more sense to teach the words individually as students work on their puzzles in the lab. You will need to decide what you believe is best for your class.

Watch the **Pair Programming - Student Video** with your students, then assign them to pairs. This should help students start off in the right direction.

Teacher Tip

Show the students the right way to help classmates:

- Don't sit in the classmate's chair
- Don't use the classmate's keyboard
- Don't touch the classmate's mouse
- Make sure the classmate can describe the solution to you out loud before you walk away

Teachers play a vital role in computer science education and supporting a collaborative and vibrant classroom environment. During online activities, the role of the teacher is primarily one of encouragement and support. Online lessons are meant to be student-centered, so teachers should avoid stepping in when students get stuck. Some ideas on how to do this are:

- Utilize pair programming whenever possible during the activity.
- Encourage students with questions/challenges to start by asking their partner.
- Unanswered questions can be escalated to a nearby group, who might already know the solution.
- Remind students to use the debugging process before you approach.
- Have students describe the problem that they're seeing. What is it supposed to do? What does it do? What does that tell you?
- Remind frustrated students that frustration is a step on the path to learning, and that persistence will pay off.
- If a student is still stuck after all of this, ask leading questions to get the student to spot an error on their own.

Wrap Up (5 - 10 min)

Journaling

Give the students a journal prompt to help them process some of the things that they encountered during the day.

Journal Prompts:

- Can you draw a sequence for getting ready to go to the computer lab?
- Draw a computer lab "Do" and a "Don't"
- Draw one of the **Feeling Faces - Emotion Images** that shows how you felt about today's lesson in the corner of your journal page.

Extension Activities

If students complete the puzzles from this lesson early, have them spend some time trying to come up with their own puzzles in their **Think Spot Journal - Reflection Journal**.

Standards Alignment

CSTA K-12 Computer Science Standards

- ▶ AP - Algorithms & Programming



This curriculum is available under a Creative Commons License (CC BY-NC-SA 4.0).

If you are interested in licensing Code.org materials for commercial purposes, **contact us**.

Lesson 3: Your Digital Footprint

Common Sense Education | Unplugged

Overview

In collaboration with **Common Sense Education - Website**, this lesson helps students learn about the similarities of staying safe in the real world and when visiting websites. Students will also learn that the information they put online leaves a digital footprint or “trail.” This trail can be big or small, helpful or hurtful, depending on how they manage it.

Purpose

Common Sense Education has created this lesson to teach kids the importance of understanding the permanence of something posted on the internet. By relating footprints on a map to what a student might post online, students will make important connections between being tracked by a physical footprint on a path and being tracked based on information posted online.

Agenda

Warm Up (20 min)

Vocabulary

Pause and Think

Main Activity (20 min)

Follow the Digital Trail - Worksheet

Wrap Up (15 min)

Flash Chat: What did we learn?

Journaling

Assessment (5 min)

Digital Footprint - Assessment

Extended Learning

View on Code Studio

Objectives

Students will be able to:

- Understand that being safe when they visit websites is similar to staying safe in real life.
- Learn to recognize websites that are safe for them to visit.
- Recognize if they should ask an adult they trust before they visit a particular website.
- Explore what information is appropriate to be put online.

Preparation

Watch this **Your Digital Footprint - Teacher Video**.

Prepare to show **Your Digital Footprint - Lesson Video**.

(Optional) Prepare to show **Pause and Think Online - Video**.

Common Sense Education's **Follow the Digital Trail - Worksheet** game.

Print one **Animal Tracks** chart (page 7) for each student.

Print one **Digital Footprint - Assessment** for each student.

Review **CSF Digital Citizenship - Resource List** for more online safety content.

Links

Heads Up! Please make a copy of any documents you plan to share with students.

For the Teacher

- **Your Digital Footprint - Teacher Video**
- **Your Digital Footprint - Lesson Video**
- **Follow the Digital Trail - Worksheet**
- **Digital Footprint - Assessment**
- **Digital Footprint - Assessment Answer Key**
- **Common Sense Education - Website**

- **CSF Digital Citizenship** - Resource List

For the Students

- **Feeling Faces** - Emotion Images
- **Think Spot Journal** - Reflection Journal

Vocabulary

- **Digital Footprint** - The collected information about an individual across multiple websites on the Internet.

Teaching Guide

Warm Up (20 min)

Vocabulary

This lesson has one new and important phrase:

- **Digital Footprint** - Say it with me: Dih-jih-tal Foot-print

The information about someone on the internet.

Pause and Think

- Ask What does it mean to be safe?
- When you walk down the street or play in your neighborhood without a trusted adult there, how do you stay safe?
- Tell students that just as they should stay safe in the real world, they should stay safe when they go into the online world (visiting websites). Make parallels between the answers students gave you about their neighborhood and the online world.

Play the **Your Digital Footprint - Lesson Video**.

- Introduce the idea that there are three different kinds of websites that students may have the opportunity to visit.
 - Green: A “green” website is:
 - A good site for kids your age to visit
 - Fun, with things for you to do and see
 - Has appropriate words
 - Doesn’t let you talk to people you don’t know
 - Yellow: A “yellow” website is:
 - A site you are not sure is right for you
 - One that asks for information such as who you are, where you live, your phone number or email address, etc.
 - A place where you are allowed to communicate freely with others
 - Red: A “red” website is:
 - A site that is not right for you
 - A place you might have gone to by accident
 - Filled with things that are for older kids or adults
- Discuss examples of each of these kinds of sites.

Lesson Tip

If you have access to a computer, feel free to navigate to sites that might showcase each of these types (using extreme caution with your RED selection).

Now, let's see what we can do to keep ourselves safe.

Main Activity (20 min)



Follow the Digital Trail - Worksheet

- Peruse the **Follow the Digital Trail - Worksheet** lesson on the Common Sense Education webpage.
- Give each student an **Animal Tracks Chart** (page 7).

	Mizzle the Mouse	Electra the Elephant
Whose full name do you know?		
Whose house could you find?		
Whose birth date do you know?		
Whose user name and password do you know?		
Who let out a secret on the internet?		
Which animal can you describe better from his or her photo?		

Directions:

- Place the **Digital Trail Squares** on the ground, face down, in two different trails, keeping Mizzle the Mouse and Electra the Elephant’s trails separate from one another.
- Share the stories of Mizzle and Electra. These animals decided it would be fun to put some information about themselves online. They went onto **www.wildkingdom.com** and posted information. The only problem is that they forgot to ask their parents if it was okay first.
- Explain to students that they are from the “Things Big and Small” Detective Agency. A hunter has hired them to find out as much as possible about Mizzle the Mouse and Electra the Elephant. The more the detectives learn, the better for their plan to take over the animal kingdom.
- Divide students into groups of four. Tell them that each group should have a detective that will keep detailed notes.
- Invite students to go on a hunt for information. Let them know that the information that Mizzle and Electra post can be seen by anyone, including the detectives. Each group should follow the digital trail of both animals, starting with the mouse and then the elephant. Stagger the groups so they are on the trail at slightly different times. Students should fill out their handout as they go.

Lesson Tip

If your students have trouble writing, feel free to do this activity as a group and have students raise their hand when they find clues. This will allow you (or a teacher aide) to help communicate and record the information being shared.

For more in-depth modules, you can find additions to this curriculum at the **Common Sense Education - Website** page on Scope and Sequence.

Wrap Up (15 min)

Flash Chat: What did we learn?

- Who can the detectives find out more about, and why?
- Which animal has a bigger digital footprint?
- Mizzle says some interesting things about himself on the Internet. What are they?
- Is there anything that Electra posted on the Internet that could become a problem for her? If so, what and why?

Lesson Tip

Flash Chat questions are intended to spark big-picture thinking about how the lesson relates to the greater world and the students' greater future. Use your knowledge of your classroom to decide if you want to discuss these as a class, in groups, or with an elbow-partner.

Take the time to discuss what is appropriate information to share on the Internet, and what is not:

Appropriate	Not Appropriate
Interests	Address
Hobbies	Full Name
First Name	Information that would hurt others

Journaling

Having students write about what they learned, why it's useful, and how they feel about it can help solidify any knowledge they obtained today and build a review sheet for them to look to in the future.

Journal Prompts:

- What was today's lesson about?
- Draw one of the **Feeling Faces - Emotion Images** that shows how you felt about today's lesson in the corner of your journal page.
- Draw some things that you should never talk to a stranger about on the internet. For example, draw your house to represent your address, draw your school, or draw your family.

Assessment (5 min)

Digital Footprint - Assessment

Hand out the assessment worksheet and allow students to complete the activity independently after the instructions have been well explained. This should feel familiar, thanks to the previous activities.

Extended Learning

Use these activities to enhance student learning. They can be used as outside of class activities or other enrichment.

Common Sense Education

- Visit **Common Sense Education - Website** to learn more about how you can keep your students safe in this digital age.

Standards Alignment

CSTA K-12 Computer Science Standards

- ▶ **IC** - Impacts of Computing



This curriculum is available under a Creative Commons License (CC BY-NC-SA 4.0).

If you are interested in licensing Code.org materials for commercial purposes, **contact us**.

Lesson 4: My Robotic Friends Jr.

Algorithms | Debugging | Unplugged

Overview

Using a set of symbols in place of code, students will design algorithms to instruct a "robot" to stack cups in different patterns. Students will take turns participating as the robot, responding only to the algorithm defined by their peers. This segment teaches students the connection between symbols and actions, the difference between an algorithm and a program, and the valuable skill of debugging.

Purpose

This unplugged lesson brings the class together as a team with a simple task to complete: get a "robot" to stack cups in a specific design. This activity lays the groundwork for the programming that students will do throughout the course as they learn the importance of defining a clearly communicated algorithm.

Agenda

Warm Up (5 min)

Talking to Robots

Activity (30 min)

Introduction and Modeling

Handy Rules:

Differentiation Options:

Programming Your Robots

Wrap Up (10 min)

Journaling

[View on Code Studio](#)

Objectives

Students will be able to:

- Attend to precision when creating instructions
- Identify and address bugs or errors in sequenced instructions

Preparation

Watch the **My Robotic Friends - Teacher Video**.

(Optional) Print out one **My Robotic Friends Symbol Key (Course B) - Key** per group or 2-3. Alternatively, find a place to display this information where students can reference throughout the lesson.

Prepare a stack of 10 disposable cups per group of 2-3 students, OR

(Optional) print and cut out **Paper Trapezoid Template - Manipulatives** for each group if your class is not going to use cups.

Print out one set of **My Robotic Friends Cup Stack Pack (Course B) - Image Pack** per group.

Make sure each student has a **Think Spot Journal - Reflection Journal**.

Links

Heads Up! Please make a copy of any documents you plan to share with students.

For the Teacher

- **My Robotic Friends** - Teacher Video

For the Students

- **My Robotic Friends Cup Spacing Template** - Template
- **Feeling Faces** - Emotion Images
- **My Robotic Friends Symbol Key (Course B)** - Key

- **My Robotic Friends** - Unplugged Video ([download](#))
- **My Robotic Friends Cup Stack Pack (Course B)** - Image Pack
- **Paper Trapezoid Template** - Manipulatives
- **Think Spot Journal** - Reflection Journal

Vocabulary

- **Algorithm** - A precise sequence of instructions for processes that can be executed by a computer
- **Bug** - Part of a program that does not work correctly.
- **Debugging** - Finding and fixing problems in an algorithm or program.
- **Program** - An algorithm that has been coded into something that can be run by a machine.

Teaching Guide

Warm Up (5 min)

Talking to Robots

Display: Watch one of the videos below to give students context for the types of things that robots can do:

- **Asimo by Honda** (3:58)
- **Egg drawing robot** (3:15)
- **Dancing Lego Robot** (1:35)

🗣️ **Discuss:** Refer to the video that you chose and ask students how they think that the robot knew what to do. Does a robot really “understand” what you say? Is it worried about getting in trouble if it doesn't do what it's told?

Say: Robots can only do what they've been told to do, but we don't just tell them using words. In order to do something, a robot needs to have a list of steps that it can read. Today, we are going to learn what it takes to make that happen.

Activity (30 min)

Discussion

The goal of this quick discussion is to call out that while robots may seem to behave like people, they're actually responding only to their programming. Students will likely refer to robots from movies and TV that behave more like humans. Push them to consider robots that they've seen or heard of in real life, like Roombas, or even digital assistants like Amazon Alexa.



Unplugged

My Robotic Friends

Symbol Key (Course B)



Pick Up Cup



Put Down Cup



Step Forward



Step Backward

Introduction and Modeling

Set Up: Have stacks of cups or cut paper trapezoids available for groups.

Display: Display the **My Robotic Friends Symbol Key (Course B) - Key** or write the allowed actions on the board - make sure these are in a place where they can be seen for the whole activity. Explain to the class that these will be the only four actions that they can use for this exercise. For this task, they will instruct their “robot” friend to build a specific cup stack using only the commands listed on the key.

Model: In order to explain how the instructions are intended to work, model for the class how to create and follow an algorithm for replicating a simple pattern. Place a single stack of cups in front of you to start.

Display: Hold up the pattern you plan to model. A simple three cup pattern is a great place to start.

Handy Rules:

- **Up** means that the cup automatically goes up as high as it needs to
- **Down** means that it automatically goes down until it lands on something
 - The hand automatically returns to cup stack after setting down a cup
- **Forward** means the robot moves one step (1/2 cup width) forward
- **Backward** means the robot moves one step (1/2 cup width) Backward
 - Note: Students may not use backward at this age unless they want to build the cup stacks in reverse (which is also okay)
- Programmers are not allowed to talk when the robot is working. This includes blurting out answers or pointing out when the robot has done something wrong
- Programmers should raise their hand if they see a bug

Differentiation Options:

Simplify: Does this all feel a little complicated for your students?

Don't forget to model this in front of the class until students understand all of the rules. If it's still confusing, try running this whole activity together as a classroom using volunteers as robots, instead of breaking up into groups!

Intensify: Are your students more advanced? Do you want this lesson to relate more closely to the online puzzles? Here are some modifications that you can make:

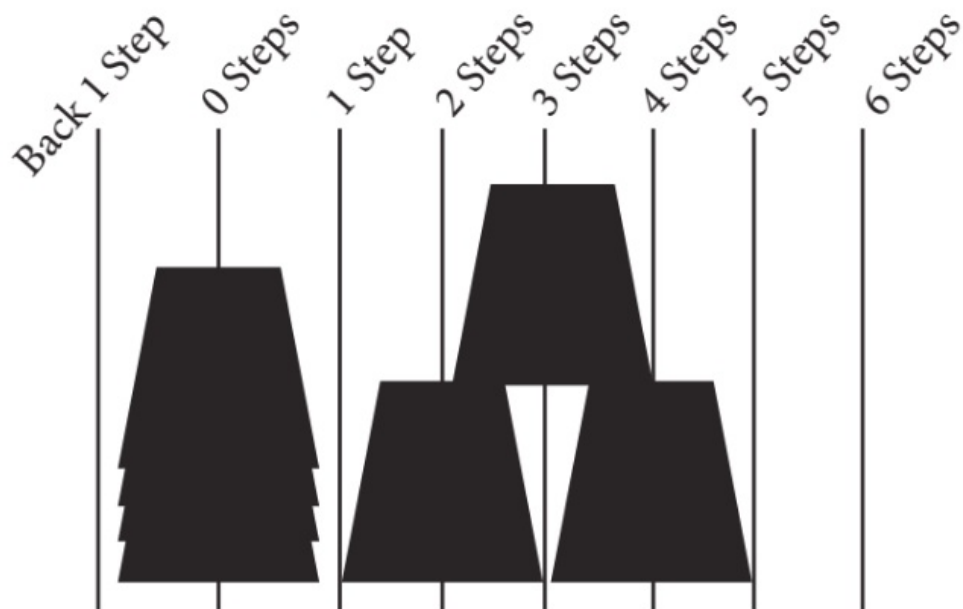
- One arrow corresponds to one movement
 - When a cup is removed from the stack, it returns to table-level before moving
 - Students need to use multiple "up" arrows to lift the cup multiple levels
 - Students need to use multiple "down" arrows to lower the cups multiple levels
 - Students need to use the "back" arrows to get back to the cup stack





Prompt: Ask the class what the first instruction should be, using **only the four instructions allowed**. The first move should be to "pick up cup." If students suggest something else from the list, perform that action and allow them to see their error. If they suggest something not from the list, make a clear malfunction reaction and let them know that the command is not understood.

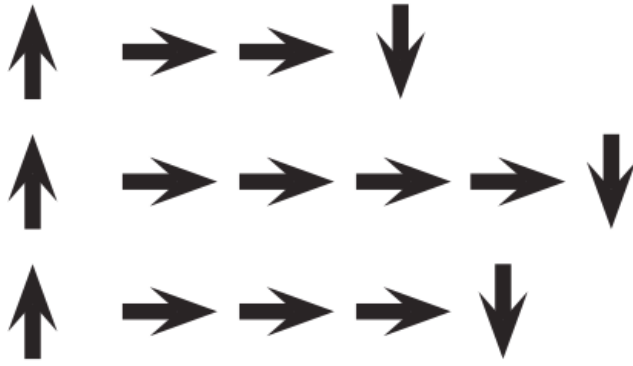
With cup in hand, ask the class to continue giving you instructions until the first cup is placed. This is a great place to clarify that "step forward" and "step backward" each imply moving half a cup width. See the image below for reference.



Continue asking for instructions from the classroom until you have completed the entire design.

Once your stack is complete, point out that they just gave you a list of steps for completing a task. That's an algorithm. Algorithms are great for sharing ideas, but spelling them out word by word can take a long time. That's what the symbols are for! When you change an algorithm into symbols that a robot (or computer) understands, that's called programming.

Ask the class to help you write the "program" for that first move by changing the text into an arrow. Then work with them to write down the rest of the moves necessary to complete the pattern. Depending on the confidence of your students, you might switch back and forth frequently between acting as the "robot" and writing down the code, or you might push them to write the whole program before you will implement it. One possible solution looks like this:



Volunteer: Once the class has completed the model program, ask one of the students to come up and act as the "robot" to ensure that the program really works. Encourage them to say the instructions out loud as they "run" the code.

Programming Your Robots

Group: Place students into groups of 4. Each group should then further break down into two pairs - each pair will develop their own program to be "run" by the other pair.

Distribute: Give each group one stack of cups or paper cutouts.

Display: Show **My Robotic Friends Cup Stack Pack (Course B) - Image Pack** to the class or hand out individual copies for groups to use. Have each pair (not group) choose which idea they would like their robots to do. Try to push for an easier idea for the first time, then have them choose a more complex design later on. Encourage pairs to keep their choice secret from the other half of their group.

Discuss: Give each pair time to discuss how the stack should be built, using only the provided symbols. Make sure each group writes down the "program" somewhere for the "robot" to read later.

Do: Once both of the group's pairs have completed their programs, they can take turns being "robots" for each other by following the instructions the other pair wrote. Encourage students to watch their "robot" closely to ensure that they are following instructions. If a student sees a bug and raises their hand, have the robot finish the instructions to the best of their ability. Afterward, have the students discuss the potential bug and come up with solutions. Continue repeating until the stack is built properly.

Circulate: Look for groups who are trying to take shortcuts by adding extra things (like numbers) to their code. Praise them for their ingenuity, but remind them that for this exercise, the robots do not understand **anything** but the provided symbols. If you like, you can hint that they should save their brilliant solution for the next time they play this game, since they might get the chance to use their invention soon!

Iterate: Depending on the time available, mix up the pairs and give them a chance to do a different pattern. Each time groups repeat the process, encourage them to choose a more challenging pattern.

Discuss: After everyone has had a chance to be the robot, bring the class back together to discuss their experience. In particular, discuss as a class:

- What was the most difficult part of coming up with the instructions?
- Did anyone find a bug in your instructions once your robot started following them?
 - What was the bug?
 - Why do you think you didn't notice it when writing the program?
- When you were the robot, what was the hardest part of following the instructions you were given?

Discussion

Sense making: The goal of this discussion is to give students space to make sense of their experience both as robot and programmer. The questions are intentionally broad, but designed to get students thinking about the challenges of writing a clear program and the constraints of a robot or computer in interpreting your instructions.

Wrap Up (10 min)

Journaling

Having students write about what they learned, why it's useful, and how they feel about it can help solidify any knowledge they obtained today and build a review sheet for them to look to in the future.

Journal Prompts:

- Draw one of the **Feeling Faces - Emotion Images** that shows how you felt about today's lesson in the corner of your journal page.
- Draw your own stack of cups that you would like to see a robot build.
- Can you create a program for that cup stack?

Standards Alignment

CSTA K-12 Computer Science Standards

- ▶ **AP** - Algorithms & Programming



This curriculum is available under a Creative Commons License (CC BY-NC-SA 4.0).

If you are interested in licensing Code.org materials for commercial purposes, **contact us**.

Lesson 5: Programming with Scrat

Algorithms | Debugging | Program | Programming

Overview

Using characters from the Ice Age, students will develop sequential algorithms to move Scrat from one side of a maze to the acorn at the other side. To do this they will stack code blocks together in a linear sequence, making them move straight, turn left, or turn right.

Purpose

In this lesson, students will develop programming and debugging skills on a computer platform. The block-based format of these puzzles help students learn about sequence and concepts, without having to worry about perfecting syntax.

Agenda

Warm Up: The Unplugged Foundation (3 min)

Review Unplugged Activity

Bridging Activity - Choose One

1) Unplugged Activity Using Paper Blocks

2) Online Activity Using Unplugged Arrows

Online Foundation: Preview Programming in Maze (3 min)

Main Activity (30 min)

Course B Online Puzzles - Website

Wrap Up (5 - 10 min)

Journaling

Extended Learning

View on Code Studio

Objectives

Students will be able to:

- Construct a program by reorganizing sequential movements
- Build a computer program from a set of written instructions
- Choose appropriate debugging practices when solving problems

Preparation

Play through the **Course B Online Puzzles - Website** in stage 5 to find any potential problem areas for your class.

(Optional) Pick a couple of puzzles to do as a group with your class.

Make sure each student has a **Think Spot Journal - Reflection Journal**.

Links

Heads Up! Please make a copy of any documents you plan to share with students.

For the Teacher

- **Course B Online Puzzles 2018** - Website
- **CS Fundamentals Main Activity Tips** - Lesson Recommendations

For the Students

- **Feeling Faces** - Emotion Images
- **My Robotic Friends Cup Stack Pack (Course B)** - Image Pack
- **My Robotic Friends Symbol Key (Course B)** - Key
- **Unplugged Blockly Blocks (Grades K-1)** - Manipulatives
- **Think Spot Journal** - Reflection Journal

Vocabulary

- **Algorithm** - A list of steps to finish a task.
- **Bug** - Part of a program that does not work correctly.
- **Debugging** - Finding and fixing problems in an algorithm or program.
- **Program** - An algorithm that has been coded into something that can be run by a machine.
- **Programming** - The art of creating a program.

Teaching Guide

Warm Up: The Unplugged Foundation (3 min)

🔗 Review Unplugged Activity

This lesson relies on many of the unplugged ideas that students have learned in the weeks leading up to this first online activity. It is important that you bring those concepts (such as persistence, debugging, algorithms, and programs) around full-circle so that your class can benefit from them in their online work as well.

Display: Show students a cup stack from the "My Robotic Friends" exercise that they completed in the lessons prior to this one.

Discuss: Ask students to recall the symbols used in "My Robotic Friends."

- What happens when the robot reads the "North" arrow?
- How about the "East" arrow?

Blend in some context from the story "Unspotted Bugs" as well.

- What would happen if we made a mistake when programming the Robot? What if there was a "bug" in our program? Would we throw the whole thing away and start over?

Encourage students to think about the debugging tips:

- Was everything right at the first step?
- How about the second?
- Where did it go wrong?

Transition: Once you are satisfied that your students remember "My Robotic Friends" and "Unspotted Bugs", you can move into the Bridging Activity.

Bridging Activity - Choose One

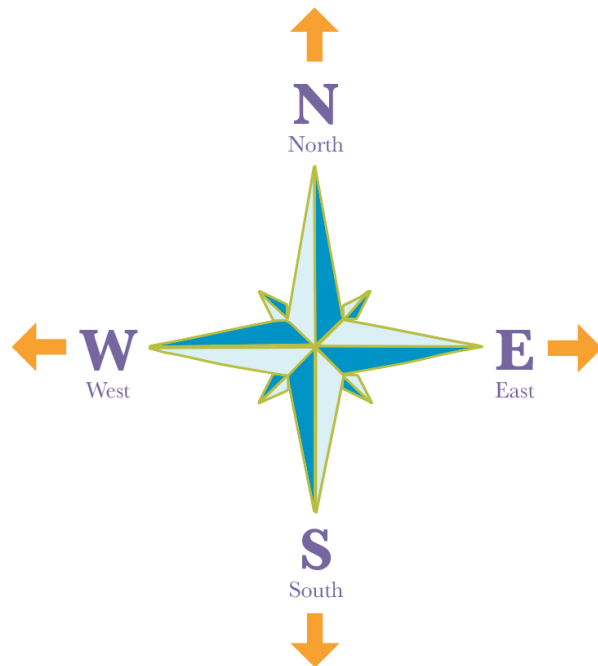
To connect the unplugged lesson with the upcoming online lesson, choose **one** of the following activities to do with your class.

1) Unplugged Activity Using Paper Blocks

Model: Select a pattern from **My Robotic Friends Cup Stack Pack (Course B) - Image Pack** from the My Robotic Friends unplugged activity (if you just modeled an image to review, feel free to keep that one for this portion of the exercise.) Using movement pieces from the **Unplugged Blockly Blocks (Grades K-1) - Manipulatives**, show students how you would code this structure in this new way.

🔗 Teaching Tip

If your class has already learned cardinal directions, then changing "Up" and "Down" to "North" and "South" shouldn't be a problem. If they have not, we have provided a handy worksheet with the Code.org Compass Rose that you can use to get students onboard. This conversion will come in handy for nearly all of the online puzzles aimed at kindergarten and first grade.



Let students know that they will see those letters in their online programs next to the direction arrows.

Pair/Think: Next, choose another pattern and have the students program what blocks a "robot" would need to read to get the correct stacking of those cups.

Make sure that they understand that the blocks need to go from top to bottom and they all need to touch!

Share: Have the students check each other's answers and resolve any questions or bugs that may come up.

2) Online Activity Using Unplugged Arrows

Model: Pull a puzzle from the corresponding online levels. We recommend Lesson 5, Puzzle 5. Show students how to get Scrat to the acorn using the **My Robotic Friends Symbol Key (Course B) - Key**. It can be helpful to rename the arrows "North", "South", "East", and "West". Once you have a program, trace it with your finger (or a pointer) and show how Scrat will travel when the program is run.

Pair/Think: Next, move back to an easier puzzle, like Lesson 5, Puzzle 4, and have students try writing programs (using arrows) on their own.

Share: Encourage students to share their programs with other groups and see if they came up with solutions that are the same or different. Can anyone come up with another way of getting Scrat to the acorn?

Online Foundation: Preview Programming in Maze (3 min)

To finish the connection, preview an online puzzle (or two) as a class.

Model: Reveal an entire online puzzle from the progression to come. We recommend Lesson 5, Puzzle 5. Point out the "Play Area" with Scrat, as well as the "Work Space" with the Blockly code. Explain that this Blockly code is now the language that students will be using to get Scrat to the acorn. Do they see any similarities to the exercise that they just did? What are the big differences?

Work with your class to drag code into the workspace in such a way that Scrat (eventually) gets to the acorn.

Transition: Students should now be ready to transition to computers to complete online puzzles on their own.

Main Activity (30 min)

If you are looking for some extra puzzles to cover with your class, here are some "prediction" puzzles that will allow you to walk through existing code with your students to predict what Scrat will do. It is a good idea to cover them together before letting students loose on their own machines.

Prediction Levels:

- **Course B, Programming in Maze #1**
- **Course B, Programming in Maze #2**

Course B Online Puzzles - Website

Circulate: Teachers play a vital role in computer science education and supporting a collaborative and vibrant classroom environment. During online activities, the role of the teacher is primarily one of encouragement and support. Online lessons are meant to be student-centered, so teachers should avoid stepping in when students get stuck. Some ideas on how to do this are:

- Utilize **Pair Programming - Student Video** whenever possible

💡 Teacher Tip:

Show the students the **right** way to help classmates by:

- Don't sit in the classmate's chair
- Don't use the classmate's keyboard
- Don't touch the classmate's mouse
- Make sure the classmate can describe the solution to you out loud before you walk away

- Encourage students with questions/challenges to start by asking their partner
- Unanswered questions can be escalated to a nearby group, who might already know the solution
- Remind students to use the debugging process before you approach
- Have students describe the problem that they're seeing. What is it supposed to do? What does it do? What does that tell you?
- Remind frustrated students that frustration is a step on the path to learning, and that persistence will pay off
- If a student is still stuck after all of this, ask leading questions to get the student to spot an error on their own

Wrap Up (5 - 10 min)

Journaling

Having students write about what they learned, why it's useful, and how they feel about it can help solidify any knowledge they obtained today and build a review sheet for them to look to in the future.

Journal Prompts:

- Draw one of the **Feeling Faces - Emotion Images** that shows how you felt about today's lesson in the corner of your journal page.
- Draw Scrat and an acorn somewhere on your paper. Can you write a program to get to get Scrat to the acorn?

Extended Learning

In small groups, let students design their own mazes on paper and challenge other students or groups to write programs to solve them. For added fun, make life-size mazes with students as Scrat and the acorn.

Standards Alignment

CSTA K-12 Computer Science Standards

- ▶ **AP** - Algorithms & Programming



This curriculum is available under a Creative Commons License (CC BY-NC-SA 4.0).

If you are interested in licensing Code.org materials for commercial purposes, **contact us**.

Lesson 6: It's Great to Create and Play Fair

Overview

Loaned to Computer Science Fundamentals by the team over at Copyright and Creativity, this lesson exists to help students understand the creative process of sharing and inspiration. Along with that comes the promotion of creative expression and the need to be fair with creative work.

Purpose

Students will soon be creating projects to share and most of these projects will contain either code or imagery that students did not create themselves. This lesson is here to show students the proper way to handle the use of content that is not their own.

Agenda

Warm-Up (Optional) (15 min)

Create Your Own Superhero! (15 min)

Wrap-Up

Journaling

Extended Learning

[View on Code Studio](#)

Objectives

Students will be able to:

- Explain why it is not okay to claim that someone else's work is your own.
- Create original art for the purpose of empathizing with other creators.

Preparation

- Review the original **It's Great to Create and Play Fair - Lesson Plan** lesson plan
- Watch the **It's Great to Create and Play Fair (Video) - Video**
- Prepare drawing/painting materials for main activity

Links

Heads Up! Please make a copy of any documents you plan to share with students.

For the Teacher

- **It's Great to Create and Play Fair - Lesson Plan**

For the Students

- **It's Great to Create and Play Fair (Video) - Video**

Teaching Guide

Warm-Up (Optional) (15 min)

Watch: Play the **It's Great to Create and Play Fair (Video) - Video** with your students. You can play it several times throughout this lesson.

The following discussions are designed to create context, help students engage with the topic, and prepare them for the lesson discussion.

Character Sketch

Ask students to help you create a character sketch about one of the characters in the video.

Say: We're going to do a character sketch where we decide who this person is.

Discuss: Which one of the characters should we use? Let's vote. [Take a quick vote.]

💡 Write the character sketch on the board as students contribute ideas. Prompt with questions:

- Who is this character?
- What is his/her name?
- Who are his/her friends? How long have they known each other?
- Who are the people in his/her family? What are they like?
- What is his/her backstory?
- Where does he/she live?
- Has he/she lived there all his/her life or has he/she moved from somewhere else?
- What exciting thing might have happened to him/her back in kindergarten, first grade?
- What does he/she look forward to?
- What is he/she afraid of?

💡 Teaching Tip

Encourage students to be as creative as they can. There are no wrong answers. Give this character a life of his/her own.

Create Your Own Superhero! (15 min)

Display: If possible, show images of two or three superheroes on the board.

Discuss: Discuss the stories of these characters. What are their super powers? What do they use their powers for?

(Other ways to tie the activity into the lesson: Have students classify superpowers into like groups, list similarities and differences between superheroes, or discuss the character traits and back stories of the superheroes.)

Activity: After discussion, ask your students to draw their own superhero character.

Discuss: When they are finished, discuss what they created:

- What did you create?
- What inspired your character?
- What makes your superhero different from or better than any other?
- How is yours similar to other superheroes?
- Have you ever seen a movie or played a video game that made you want to make something?

Say: I'm going to play a short video. As you watch the video, think about how you would feel if this situation happened to you. What would you do? **Watch:** Play the **It's Great to Create and Play Fair (Video) - Video** with your students.

Discuss: It's great to create. And, it's great to recognize how others' creations inspire our new creations. We want to be fair when we're using each other's creative work.

- What did you think about that situation? What was going on?
- These students combined their work to make something new — the t-shirt.
- What would you do in that situation?
- Did these friends treat each other fairly?
- What if she had wanted to add a ballerina tutu or big boots or glasses to the dragon?
- Do you think the boy would have liked that?
- Do you think that would be fair?
- How would you have felt?

Wrap-Up

Creating new things is fun — art, music, movies, paper creations, structures, even buildings! It's great to create and share and be inspired — as long as we respect each other as artists and play fair.

Journaling

Give the students a journal prompt to help them process some of the things that they encountered during the day. You can choose one of the prompts below, or make up your own.

Journal Prompts:

- Draw a feeling face in the corner of your journal page
- Think of a superhero that one of your classmates made. Can you draw your own picture of that superhero? Give proper credit to your classmate as the creator of the original version.

Extended Learning

Please be sure to visit [Copyright & Creativity](#) to find more lessons on digital sharing and creative rights.

Standards Alignment

CSTA K-12 Computer Science Standards

- ▶ AP - Algorithms & Programming



This curriculum is available under a Creative Commons License (CC BY-NC-SA 4.0).

If you are interested in licensing Code.org materials for commercial purposes, **contact us**.

Lesson 7: Programming with Rey and BB-8

Programming | Maze

Overview

In this lesson, students will use their newfound programming skills in more complicated ways to navigate a tricky course with BB-8.

Purpose

With transfer of knowledge in mind, this lesson gives students a new environment to practice the skills that they have been cultivating. Star Wars fans will jump for joy when they see these puzzles. Each puzzle in this series has been added to provide a deeper understanding of the basic concepts that they will be using throughout the rest of this course.

Agenda

Warm Up (15 min)

Introduction

Main Activity (30 min)

Course B Online Puzzles - Website

Wrap Up (15 min)

Journaling

[View on Code Studio](#)

Objectives

Students will be able to:

- Sequence commands in a logical order.
- Recognize problems or "bugs" in a program and develop a plan to resolve the issues.

Preparation

Play through the **Course B Online Puzzles - Website** corresponding with this lesson to find any potential problem areas for your class.

Review **CS Fundamentals Main Activity Tips - Lesson Recommendations**.

Make sure every student has a **Think Spot Journal - Reflection Journal**.

Links

Heads Up! Please make a copy of any documents you plan to share with students.

For the Teacher

- **Course B Online Puzzles 2018** - Website
- **CS Fundamentals Main Activity Tips** - Lesson Recommendations

For the Students

- **Feeling Faces** - Emotion Images
- **Think Spot Journal** - Reflection Journal

Vocabulary

- **Algorithm** - A list of steps to finish a task.
- **Bug** - Part of a program that does not work correctly.
- **Debugging** - Finding and fixing problems in an algorithm or program.
- **Program** - An algorithm that has been coded into something that can be run by a machine.
- **Programming** - The art of creating a

program.

Teaching Guide

Warm Up (15 min)

Introduction

Ask the students how they felt about the last lesson.

- Which puzzles were too hard or too easy?
- Which puzzles were frustrating or a lot of fun?
- If they were to teach the lesson to a friend, which part of the lesson would they want to review?

Use these questions to form a brief review of programming and debugging. If you think the class could benefit from it, you can go over the vocabulary words and definitions from the last lesson.

If you feel comfortable, also give a brief introduction to BB-8 from Star Wars. Many students may already be familiar with the lovable robot, but the introduction will surely build excitement.

Main Activity (30 min)

Course B Online Puzzles - Website

As we mentioned in the last lesson, we highly recommend viewing and using **Pair Programming - Student Video** as a class. Pair programming stimulates a discussion that can answer questions, review basic concepts, and build confidence with the subject.

Wrap Up (15 min)

Journaling

Having students write about what they learned, why it's useful, and how they feel about it can help solidify any knowledge they obtained today and build a review sheet for them to look to in the future.

Journal Prompts:

- What was today's lesson about?
- Draw one of the **Feeling Faces - Emotion Images** that shows how you felt about today's lesson in the corner of your journal page.
- Draw a picture of BB-8 you guided through the maze today and add a list of the commands that you used.

Standards Alignment

CSTA K-12 Computer Science Standards

- ▶ AP - Algorithms & Programming



This curriculum is available under a Creative Commons License (CC BY-NC-SA 4.0).

If you are interested in licensing Code.org materials for commercial purposes, **contact us**.

Lesson 8: My Loopy Robotic Friends Jr.

Unplugged | Loop | Repeat

Overview

Building on the initial "My Robotic Friends" activity, students tackle larger and more complicated designs. In order to program their "robots" to complete these bigger designs, students will need to identify repeated patterns in their instructions that could be replaced with a loop.

Purpose

This lesson serves as a reintroduction to loops, using the now familiar set of "robot" programming instructions. Students will develop critical thinking skills by looking for patterns of repetition in the movements of classmates and determining how to simplify those repeated patterns using loops.

Agenda

Warm Up (10 min)

My Robotic Friends Review

Activity (30 min)

Introduction and Modeling Looping Your Robots

Wrap Up (5 min)

Extension Activities

View on Code Studio

Objectives

Students will be able to:

- Identify repeated patterns in code that could be replaced with a loop
- Write instructions that use loops to repeat patterns

Preparation

Make sure each student has a **Think Spot Journal - Reflection Journal**

Watch the **My Loopy Robotic Friends - Teacher Video**

Prepare a stack of 20 paper cups for each group of 4 students

(Optional) Print out one **My Loopy Robotic Friends Cup Stack (Course B) - Image Pack** per group of 4 students

OR

Display the **My Robotic Friends Symbol Key (Course B) - Key** where students can reference throughout the lesson.

Print and cut out **Paper Trapezoid Template - Manipulatives** for each group if your class is not going to use cups.

Print out one set of **My Loopy Robotic Friends Cup Stack (Course B) - Image Pack** per group.

Links

Heads Up! Please make a copy of any documents you plan to share with students.

For the Teacher

- **My Loopy Robotic Friends - Teacher Video**

For the Students

- **My Robotic Friends Cup Spacing**

- **Template** - Template
- **My Robotic Friends Symbol Key (Course B)** - Key
- **Feeling Faces** - Emotion Images
- **Paper Trapezoid Template** - Manipulatives
- **My Loopy Robotic Friends Cup Stack (Course B)** - Image Pack

Vocabulary

- **Loop** - The action of doing something over and over again.
- **Repeat** - Do something again

Teaching Guide

Warm Up (10 min)

My Robotic Friends Review



U
Unplugged

My Robotic Friends
Symbol Key (Course B)

C	O
D	E



Pick Up Cup



Put Down Cup



Step Forward



Step Backward

Goal: This review will refresh the students' minds about how quickly programs for the "My Robotic Friends" activity can get intense.

Display: Show the **My Robotic Friends Symbol Key (Course B) - Key** that we used in My Robotic Friends. For each of the four symbols, ask students to show you what it looks like for a robot to follow that instruction.

Model: With the class together as a group, pull an easy puzzle from the "My Robotic Friends" Cup Stack Pack and program with each other as a reminder of rules and terminology.

Next, pull a puzzle that's slightly harder, but also requires a lot of steps like the one below.



Volunteer: Ask a volunteer (or a group of volunteers) to come forward to help program this one on the board. If you make them stick strictly to the “no symbols other than those on the key” rule, it will probably take a while!

Display: Now, bring up this image:



What is the reaction of the class?

Prompt: Give students the opportunity to brainstorm shorter ways to relay the code that they're about to create. (This bit can be skipped over if your students start saying things like: “Move forward 6 times.” Since that will open the discussion about how to show “six times” with symbols.)

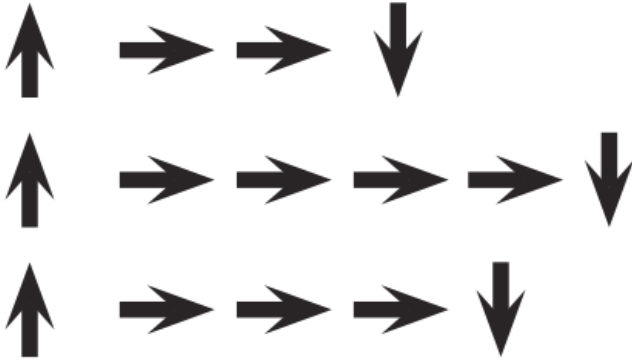
Once students have put together the idea of “repeating” code, give them the vocabulary around it. Make sure to share with them that often the terms “repeat something” and “loop something” are often used interchangeably.

Activity (30 min)

Introduction and Modeling

Set Up: Have stacks of cups or cut paper trapezoids available for groups.

Display: Take the program from one of your previous cup stacks and display it for the class, or use the one below.

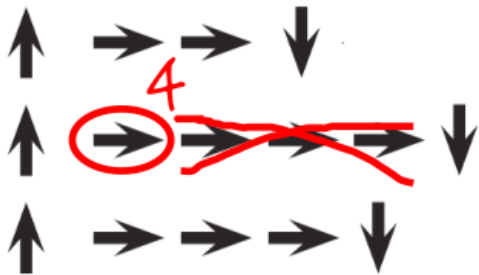


Think: Ask students to think quietly about where in this program they can find a pattern of instructions that repeat uninterrupted (one repetition after another).

Pair: Turn to a neighbor and share one of the repeating patterns you found.

Share: Ask a few students to share out the patterns they identified. Try to pull out different approaches to grouping patterns. For each pattern, ask students to identify how many times the pattern repeats.

Model: Using one of the repeating patterns that the class identified, model how to circle the instruction or pattern that repeats, write the number of loops near that circle, then cross out the rest of the arrows.



Repeat this until the entire program has been shortened, then re-write the program in a way where students can see how much more simple the resulting instructions are.

Looping Your Robots

Group: Place students into groups of 4. Each group should then further break down into two pairs - each pair will develop their own program "run" on the other pair.

Distribute: Give each group one stack of cups or paper cutouts.

Display: Show **My Loopy Robotic Friends Cup Stack (Course B) - Image Pack** to the class or hand out individual copies for groups to use. Have each pair (not group) choose which stack they would like their robot to do. Encourage pairs to select a more complicated pattern this time around.

💡 **Discuss:** Let each group discuss how the stack should be built, then instruct each group to translate the algorithm into symbols. Make sure each group writes down the symbol algorithm somewhere for the "robot" to read later. As students are working on their programs, remind them to be on the lookout for opportunities to replace a repeating pattern with a loop.

Do: When groups have finished their instructions, have each pair trade with another pair to run one another's code. Remind students to be on the lookout for bugs, but not to interrupt a robot until it's finished running the program.

Discuss: When all of the pairs have had a chance to run their programs, ask a few to share their solutions with the class. Use this opportunity to discuss how groups came up with different solutions to the same puzzle. In particular, you might ask of each program:

- How did they identify the loops?
- Are there other ways those loops could have been written?
- How much shorter is the program with loops than it would be without?
- Is the program easier to understand with loops, or written out longhand? Why?

Teaching Tip

Looking for Loops: Be sure to keep your eyes open for students using loops. Try to avoid correcting their overall algorithms or prescribing a solution, but feel free to direct students towards patterns that could be shortened by using a repeat circle.

Watch students as they run through the code. Are there any bugs? Use the debugging questions to help them find a solution.

- What does it do?
- What is it supposed to do?
- What does that tell you?
- Does it work at the first step?
- Does it work at the second step?
- Where does it stop working?

Wrap Up (5 min)

Journal Prompts:

- Draw one of the **Feeling Faces - Emotion Images** that shows how you felt about today's lesson in the corner of your journal page.
- Have the students write or draw something in their journal that will remind them later what loops are. This can come from a prompt like:
 - What does "repeat" mean to you?
 - Draw a picture of you repeating something.

Extension Activities

- Have students draw their own cup stacking creations for someone else to code.
- Provide students with algorithms that utilize repeats, then have them expand the program back out to a full step-by-step version.

Standards Alignment

CSTA K-12 Computer Science Standards

- ▶ **AP** - Algorithms & Programming



This curriculum is available under a Creative Commons License (CC BY-NC-SA 4.0).

If you are interested in licensing Code.org materials for commercial purposes, **contact us**.

Lesson 9: Loops with Scrat

Loop | Ice Age | Scrat

Overview

Building on the concept of repeating instructions from "My Loopy Robotic Friends," this stage will have students using loops to get to the acorn more efficiently on Code.org.

Purpose

In this lesson, students will be learning more about loops and how to implement them in Blockly code. Using **loops** is an important skill in programming because manually repeating commands is tedious and inefficient. With these Code.org puzzles, students will learn to add instructions to existing loops, gather repeated code into loops, and recognize patterns that need to be repeated.

Agenda

Warm Up - The Unplugged Foundation (10 min)

Review Unplugged Activity

Bridging Activity - Choose One

- 1) Unplugged Activity Using Paper Blocks
- 2) Online Activity Using Unplugged Arrows

Online Foundation: Preview Loops in Ice Age

Main Activity (30 min)

Course B Online Puzzles - Website

Wrap Up (5 - 10 min)

Journaling

Extended Learning

View on Code Studio

Objectives

Students will be able to:

- Construct a program using structures that repeat areas of code
- Improve existing code by finding areas of repetition and moving them into looping structures

Preparation

Review the previous unplugged lesson and develop questions to remind students why loops are used.

(Optional) Pick a couple of puzzles to do as a group with your class.

Review **CS Fundamentals Main Activity Tips - Lesson Recommendations**.

Gather supplies from previous My Loopy Robotic Friends to reuse for warm up

Make sure each student has a **Think Spot Journal - Reflection Journal**.

Links

Heads Up! Please make a copy of any documents you plan to share with students.

For the Teacher

- **Course B Online Puzzles 2018** - Website
- **CS Fundamentals Main Activity Tips - Lesson Recommendations**

For the Students

- **My Loopy Robotic Friends Cup Stack (Course B)** - Image Pack
- **Feeling Faces** - Emotion Images
- **My Robotic Friends Symbol Key (Course B)** - Key
- **Unplugged Blockly Blocks (Grades K-1)** - Manipulatives
- **Think Spot Journal** - Reflection Journal

Vocabulary

- **Loop** - The action of doing something over and over again.
- **Repeat** - Do something again

Teaching Guide

Warm Up - The Unplugged Foundation (10 min)

🔗 Review Unplugged Activity

This lesson relies on the concept of repeat loops that students learned in the previous unplugged activity, My Loopy Robotic Friends. It is important to bring this idea from the real world into digital form so that students understand how to use Blockly blocks to repeat a task multiple times.

Display: Show students a cup stack from the "My Loopy Robotic Friends" exercise that they completed in the lessons prior to this one.

Discuss: Ask students to recall the symbols used in "My Loopy Robotic Friends."

- What happens when "East" arrow is circled with the number 3? (It moves E 3 times)
- What is it called when we circle an arrow and add a number? (A repeat loop)

Transition: Once you are satisfied that your students remember "My Loopy Robotic Friends", you can move into the Bridging Activity.

Bridging Activity - Choose One

This activity will help bring the unplugged concepts from "My Loopy Robotic Friends" into the online world that the students are moving into. Choose **one** of the following to do with your class:

1) Unplugged Activity Using Paper Blocks

Model: Select a pattern from **My Loopy Robotic**

Friends Cup Stack (Course B) - Image Pack from the My Loopy Robotic Friends unplugged activity (if you just modeled an image to review, feel free to keep that one for this portion of the exercise.) Using movement pieces from the **Unplugged Blockly Blocks (Grades K-1) - Manipulatives**, show students how you would code this shape using Blockly repeat blocks.

Pair/Think: Next, choose another pattern and have the students program what blocks a "robot" would need to read to get the correct stacking of those cups.

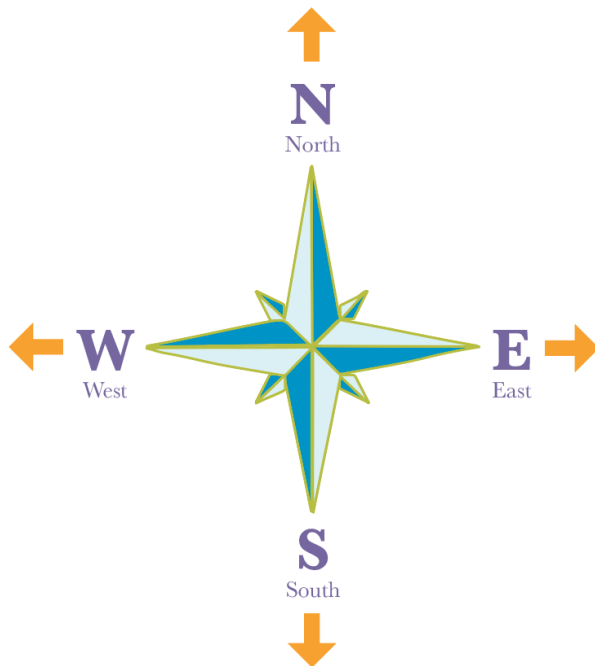
Make sure that students understand that only the repeating code needs to go inside of the pinkrepeat blocks.

Share: Have the students check each other's answers and resolve any questions or bugs that may come up.

2) Online Activity Using Unplugged Arrows

🔗 Teaching Tip

If your class has already learned cardinal directions, then changing "Up" and "Down" to "North" and "South" shouldn't be a problem. If they have not, we have provided a handy worksheet with the Code.org Compass Rose that you can use to get students onboard. This conversion will come in handy for nearly all of the online puzzles aimed at kindergarten and first grade.



Let students know that they will see those letters in their online programs next to the direction arrows.

Model: Pull a puzzle from the corresponding online levels. We recommend Lesson 8, Puzzle 4. Show students how to get Scrat to the acorn using **My Robotic Friends Symbol Key (Course B) - Key** . It can be helpful to call the arrows "North", "South", "East", and "West". Once you have a program, trace it with your finger (or a pointer) and show how Scrat will travel when the program is run.

Pair/Think: Next, move on to a puzzle that is a little more challenging, like Lesson 8, Puzzle 5, and have students try writing programs (using arrows and repeat circles) on their own.

Share: Encourage students to share their programs with other groups and see if they came up with solutions that are the same or different.

Online Foundation: Preview Loops in Ice Age

To finish the connection, preview an online puzzle (or two) as a class.

Model: Reveal an entire online puzzle from the progression to come. We recommend Lesson 8, Puzzle 5. Point out the "Play Area" with Scrat and the acorn, as well as the "Work Space" with the Blockly code. Explain that this Blockly code is now the language that the class will be using to help Scrat get to the acorn. Do students see any similarities to the exercise that they just did? What are the big differences?

Work with your class to drag code into the workspace in such a way that Scrat (eventually) gets to the acorn.

Transition: Students should now be ready to transition to computers to complete online puzzles on their own.

Main Activity (30 min)

As students work through the puzzles, see if they can figure out how many blocks they use with a loop vs. without a loop.

Course B Online Puzzles - Website

Circulate: Teachers play a vital role in computer science education and supporting a collaborative and vibrant classroom environment. During online activities, the role of the teacher is primarily one of encouragement and support. Online lessons are meant to be student-centered, so teachers should avoid stepping in when students get stuck. Some ideas on how to do this are:

- Utilize **Pair Programming - Student Video** whenever possible
- Encourage students with questions/challenges to start by asking their partner
- Unanswered questions can be escalated to a nearby group, who might already know the solution
- Remind students to use the debugging process before you approach
- Have students describe the problem that they're seeing. What is it supposed to do? What does it do? What does that tell you?
- Remind frustrated students that frustration is a step on the path to learning, and that persistence will pay off.
- If a student is still stuck after all of this, ask leading questions to get the student to spot an error on their own.

👉 Teacher Tip:

Show the students the **right** way to help classmates by:

- Don't sit in the classmate's chair
- Don't use the classmate's keyboard
- Don't touch the classmate's mouse
- Make sure the classmate can describe the solution to you out loud before you walk away

Wrap Up (5 - 10 min)

Journaling

Having students write about what they learned, why it's useful, and how they feel about it can help solidify any knowledge they obtained today and build a review sheet for them to look to in the future.

Journal Prompts:

- What was today's lesson about?
- Draw one of the **Feeling Faces - Emotion Images** that shows how you felt about today's lesson in the corner of your journal page.
- Draw Scrat and an acorn.
- Draw yourself using a loop to do an everyday activity, like brushing your teeth.

Extended Learning

So Moving

- Give the students pictures of actions or dance moves that they can do.
 - Have students arrange moves and add loops to choreograph their own dance.
- Share the dances with the rest of the class.

Connect It Back

- Find some YouTube videos of popular dances that repeat themselves.
- Can your class find the loops?
- Try the same thing with songs!

Standards Alignment

CSTA K-12 Computer Science Standards

- ▶ **AP** - Algorithms & Programming



This curriculum is available under a
Creative Commons License (CC BY-NC-SA 4.0).

If you are interested in licensing Code.org materials for commercial purposes, **contact us**.

Lesson 10: Loops with Laurel

Loop | Collector

Overview

In this lesson, students continue learning the concept of loops. Here, Laurel the Adventurer uses loops to collect treasure in open cave spaces. A new `get treasure` block is introduced to help her on her journey.

Purpose

This lesson gives students more practice with loops and encourages them to put multiple blocks inside of a `repeat` as they try to collect as much treasure as possible.

Agenda

Warm Up (10 min)

Introduction

Online Foundation: Preview Loops in Collector

Main Activity (30 min)

Course B Online Puzzles - Website

Wrap Up (5 - 10 min)

Journaling

[View on Code Studio](#)

Objectives

Students will be able to:

- Identify the benefits of using a loop structure instead of manual repetition.
- Break down a long sequence of instructions into the smallest repeatable sequence possible.

Preparation

Play through the **Course B Online**

Puzzles - Website in stage 9 to find any potential problem areas for your class.

(Optional) Pick a couple of puzzles to do as a group with your class.

Review **CS Fundamentals Main Activity Tips - Lesson Recommendations**

Make sure each student has a **Think Spot Journal - Reflection Journal**.

Links

Heads Up! Please make a copy of any documents you plan to share with students.

For the Teacher

- **Course B Online Puzzles 2018** - Website
- **CS Fundamentals Main Activity Tips** - Lesson Recommendations

For the Students

- **My Loopy Robotic Friends Cup Stack (Course B)** - Image Pack
- **Feeling Faces** - Emotion Images
- **My Robotic Friends Symbol Key (Course B)** - Key
- **Unplugged Blockly Blocks (Grades K-1)** - Manipulatives
- **Think Spot Journal** - Reflection Journal

Vocabulary

- **Loop** - The action of doing something over and over again.
- **Repeat** - Do something again

Teaching Guide

Warm Up (10 min)

Introduction

Quickly review the definition of a loop, the action of doing something over and over again.

- What are loops?
- Why do we use them?

Online Foundation: Preview Loops in Collector

To introduce Laurel the Collector, preview an online puzzle (or two) as a class.

Model: Reveal an entire online puzzle from the progression to come. We recommend Lesson 9, Puzzle 8. Do students see any similarities to the last set of exercises that they did? What are the big differences? When should the get treasure block be used?

Work with your class to drag code into the workspace in such a way that Laurel (eventually) collects all of the treasure.

Transition: Students should now be ready to transition to computers to complete online puzzles on their own.

Main Activity (30 min)

Teacher Demonstration

We've included some multiple choice prediction levels that are difficult for non-readers. These levels are optional for you to review with your class to help prepare for the puzzles to come. Alternatively, these could be used after finishing the stage as a review for the class.

Prediction Levels:

- **Course B, Loops in Collector**

Course B Online Puzzles - Website

As students work through the puzzles, see if they can figure out how many blocks they use with a loop vs. without a loop.

Wrap Up (5 - 10 min)

Journaling

Having students write about what they learned, why it's useful, and how they feel about it can help solidify any knowledge they obtained today and build a review sheet for them to look to in the future.

Journal Prompts:

- What was today's lesson about?
- Draw one of the **Feeling Faces - Emotion Images** that shows how you felt about today's lesson in the corner of your journal page.
- Draw a line of treasure that Laurel could collect.
- Draw something that uses loops.

Standards Alignment

CSTA K-12 Computer Science Standards

► **AP** - Algorithms & Programming



This curriculum is available under a
Creative Commons License (CC BY-NC-SA 4.0).

If you are interested in licensing Code.org materials for commercial purposes, **contact us**.

Lesson 11: Drawing Gardens with Loops

Loop | Artist

Overview

Returning to loops, students learn to draw images by looping simple sequences of instructions. In the previous online lesson, loops were used to traverse a maze and collect treasure. Here, students use loops to create patterns. At the end of this stage, students will be given the opportunity to create their own images using loops.

Purpose

This lesson gives a different perspective on how loops can create things in programming. Students will test their critical thinking skills by evaluating given code and determining what needs to be added in order to solve the puzzle. Students can also reflect on the inefficiency of programming without loops here because of how many blocks the program would require without the help of repeat loops.

Agenda

Warm Up (10 min)

Introduction

Main Activity (30 min)

Course B Online Puzzles - Website

Wrap Up (5 - 10 min)

Journaling

[View on Code Studio](#)

Objectives

Students will be able to:

- Count the number of times an action should be repeated and represent it as a loop.
- Decompose a shape into its largest repeatable sequence.
- Create a program that draws complex shapes by repeating simple sequences.

Preparation

- Play through the **Course B Online Puzzles - Website** before the lesson to find any potential problem areas for your class.
- Review **CS Fundamentals Main Activity Tips - Lesson Recommendations**.
- Make sure each student has a **Think Spot Journal - Reflection Journal**.

Links

Heads Up! Please make a copy of any documents you plan to share with students.

For the Teacher

- **Course B Online Puzzles 2018** - Website
- **CS Fundamentals Main Activity Tips** - Lesson Recommendations

For the Students

- **Feeling Faces** - Emotion Images
- **Think Spot Journal** - Reflection Journal

Vocabulary

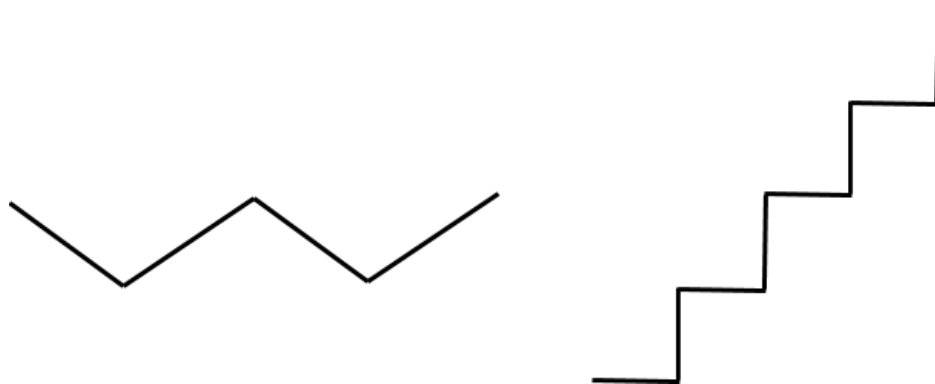
- **Loop** - The action of doing something over and over again.

Teaching Guide

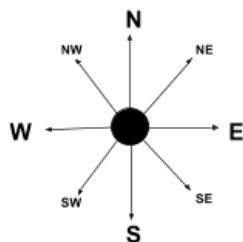
Warm Up (10 min)

Introduction

- Quickly review the definition of a loop, the action of doing something over and over again.
- Discuss different patterns like zigzags and stairsteps.
 - How would you explain to someone how to draw that pattern?
 - How could you draw this using a loop?



In the artist levels, students will be using 45 degree angles described as northwest, northeast, southwest, southeast. We recommend briefly discussing these directions with the class and drawing an image for students to refer back to.



Main Activity (30 min)

Teacher Demonstration

We've included some multiple choice prediction levels that are difficult for non-readers. These levels are optional for you to review with your class to help prepare for the puzzles to come. Alternatively, these could be used after finishing the stage as a review for the class.

Prediction Levels:

- **Course B, Loops in Artist**

Course B Online Puzzles - Website

Wrap Up (5 - 10 min)

Journaling

Teacher Tip

Remind the students to only share their work with their close friends or family. For more information watch or show the class **Pause and Think Online - Video** .

Having students write about what they learned, why it's useful, and how they feel about it can help solidify any knowledge they obtained today and build a review sheet for them to look to in the future.

Journal Prompts:

- What was today's lesson about?
- Draw one of the **Feeling Faces - Emotion Images** that shows how you felt about today's lesson in the corner of your journal page.
- Draw the patterns you made with a loop.
- Draw a pattern that you would like to make with a loop.

Standards Alignment

CSTA K-12 Computer Science Standards

- ▶ **AP** - Algorithms & Programming



This curriculum is available under a Creative Commons License (CC BY-NC-SA 4.0).

If you are interested in licensing Code.org materials for commercial purposes, **contact us**.

Lesson 12: The Big Event Jr.

Event | Unplugged

Overview

Events are a great way to add variety to a pre-written algorithm. Sometimes you want your program to be able to respond to the user exactly when the user wants it to. That is what events are for.

Purpose

Today, students will learn to distinguish events from actions. The students will see activities interrupted by having a "button" pressed on a paper remote. When seeing this **event**, the class will react with a unique action. Events are widely used in programming and should be easily recognizable after this lesson.

Agenda

Warm Up (15 min)

Vocabulary
A Series of Events

Main Activity (15 min)

The Big Event

Wrap Up (10 min)

Flash Chat: What did we learn?
Journaling

Assessment (10 min)

The Big Event - Assessment

Extended Learning

View on Code Studio

Objectives

Students will be able to:

- Repeat commands given by an instructor.
- Recognize actions of the teacher as signals to initiate commands.
- Practice differentiating pre-defined actions and event-driven ones.

Preparation

Watch the **The Big Event - Teacher Video**.

Print one **The Big Event (Courses A, B) - Worksheet**.

Print one **The Big Event - Assessment** for each student.

Make sure each student has a **Think Spot Journal - Reflection Journal**.

Links

Heads Up! Please make a copy of any documents you plan to share with students.

For the Teacher

- **The Big Event** - Unplugged Video ([download](#))
- **The Big Event** - Teacher Video
- **The Big Event (Courses A, B)** - Worksheet
- **The Big Event** - Assessment
- **The Big Event** - Assessment Answer Key

For the Students

- **Feeling Faces** - Emotion Images
- **Think Spot Journal** - Reflection Journal

Vocabulary

- **Event** - An action that causes something to happen.

Teaching Guide

Warm Up (15 min)

Vocabulary

This lesson has one new and important vocabulary word:

Event - Say it with me: E-vent

An action that causes something to happen

A Series of Events

- Prep your class to answer a question:
 - "I'm going to ask you a question. I want you to raise your hand if you want me to call on you for the answer."
 - Ask a simple question that most of your students should be able to answer, such as:
 - How many thumbs do I have?
 - What is bigger, a bird or a horse?
 - Call on a student who has their hand raised and let them give their answer.
 - Upon finishing that display, ask the class how you knew that the student wanted you to call on them.
 - Your class will likely mention the raising of the hand.
 - Explain to everyone that when students raise their hand, it is an "event" that causes you to know that they want to be called on.
- Ask the class if they can think of any other events that give signals.
 - You may need to remind them that you're not talking about an event like a birthday party or a field trip.
 - If they have trouble, you can remind them that an event is an action that causes something to happen.
 - What about an alarm clock going off? What does that make happen?
 - What about pressing "Start" on the microwave? What does that do?
 - What about pressing the power button on your tv remote?
- Today, we're going to create programs with events.

Main Activity (15 min)

The Big Event

- Do you remember helping the Red, the Angry Bird find the pig?
 - In that exercise, you knew in advance exactly where you wanted Red to end up, so you could make a program that took the bird from start to finish without any interruptions.
 - In most real programs, we can't do that because we want to have options, depending on what the user needs.
 - Say that I only want my character to move when my finger is on the screen of my phone. I would need to program the character to only move when I put my finger on the screen of my phone.
 - Putting my finger on the screen would then become an "event" that tells my character to move.

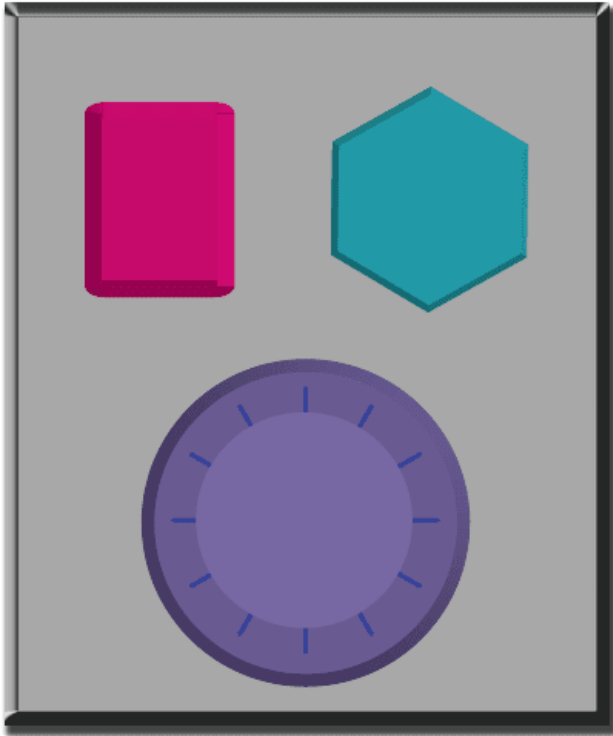
💡 Lesson Tip

If your students seem confused, talk about their favorite games and all of the ways that they let the characters know what they're supposed to do. Point out how the game would be really boring if it ran from start to finish without any events required.

In earlier lessons, we created algorithms that allowed us to control a friend or bird for several steps at a time. It was fun and useful, but what happens when you don't know everything that you want your friend to do in advance? This is where events come in!

Directions:

- Project the Event Controller onto your classroom screen.



- Decide with your class what each button does. We suggest:
 - Pink Button -> Say "Wooooo!"
 - Teal Button -> "Yeah!"
 - Purple Dial -> "Boom!"
- Practice tapping the buttons on the overhead and having your class react.
- Add some button sequences into the mix and have the students try to keep up with their sounds.
- Let your class know that every time you push a button, it is an "event" that lets them know what they are expected to do next.
- Get the class started on a planned task before interrupting them again with the buttons. We suggest:
 - Counting to 10
 - Singing "Old MacDonald"
- Once their plan is underway, interject button presses sporadically.
- Continue the blend until they understand the difference between actions that are guided by a plan and those that are event driven.

Wrap Up (10 min)

Flash Chat: What did we learn?

- Why do we need to be able to handle events in a program?
- What are some other kinds of events that you can think of?

Journaling

Journal Prompts:

- What was today's lesson about?
- Draw one of the **Feeling Faces - Emotion Images** that shows how you felt about today's lesson in the corner of your journal page.
- Draw an event that caused an action today.
- Draw an action that was caused by an event that happened today.

Assessment (10 min)

The Big Event - Assessment

- Hand out the assessment activity and allow students to complete the activity independently after the instructions have been well explained.
- This should feel familiar, thanks to the previous activities.

Extended Learning

Use these activities to enhance student learning. They can be used as outside of class activities or other enrichment.

One Person's Event is Another One's Reaction

Assign each student an event to watch out for, and an appropriate reaction to that event. Chain the actions so that each child's reaction becomes an event that triggers the reaction of another student. Keep assigning until everyone has something to do and everyone makes someone react.

Eventopalooza

Break the class up into groups. Using the Events Controller, assign each group a different reaction to the same button. Do this for all three buttons, then watch the chaos!

Standards Alignment

CSTA K-12 Computer Science Standards

- ▶ AP - Algorithms & Programming



This curriculum is available under a Creative Commons License (CC BY-NC-SA 4.0).

If you are interested in licensing Code.org materials for commercial purposes, **contact us**.

Lesson 13: A Royal Battle with Events

Event | Play Lab

Overview

In this online activity, students will have the opportunity to learn how to use events in Play Lab and apply all of the coding skills that they've learned to create an animated game. It's time to get creative and make a game in Play Lab!

Purpose

In this online activity, students will learn how to use events in Play Lab. They will start by training the knight to move when an arrow key is pressed, then end with the opportunity to showcase the rest of the skills that they learned throughout this course, including sequence and looping, as part of the final freeplay puzzle.

Agenda

Warm Up (10 min)

Introduction

Bridging Activity - Events (10 min)

Unplugged Activity Using Paper Blocks Previewing Online Puzzles as a Class

Main Activity (30 min)

Course B Online Puzzles - Website

Wrap Up (5 - 10 min)

Journaling

Extended Learning

View on Code Studio

Objectives

Students will be able to:

- Identify actions that correlate to input events.
- Create an animated, interactive story using sequences and event-handlers.
- Share a creative artifact with other students.

Preparation

- Play through the **Course B Online Puzzles - Website** in stage 12 to find any potential problem areas for your class.
- (Optional) Pick a couple of puzzles to do as a group with your class.
- Review **CS Fundamentals Main Activity Tips - Lesson Recommendations**.
- Make sure every student has a **Think Spot Journal - Reflection Journal**.

Links

Heads Up! Please make a copy of any documents you plan to share with students.

For the Teacher

- **Course B Online Puzzles 2018** - Website
- **CS Fundamentals Main Activity Tips** - Lesson Recommendations

For the Students

- **Feeling Faces** - Emotion Images
- **Unplugged Blockly Blocks (Grades K-1)** - Manipulatives
- **Think Spot Journal** - Reflection Journal

Vocabulary

- **Event** - An action that causes something to happen.

Teaching Guide

Warm Up (10 min)

Introduction

Review "The Big Event" activity with students:

- What did we "program" the button events to do?

Now we're going to add events to our code. Specifically, we're going to have an event for when two characters touch each other.

- When have you seen two characters touch each other as an event in games?

Bridging Activity - Events (10 min)

This activity will help bring the unplugged concepts from "The Big Event" into the online world that the students are moving into. Choose **one** of the following to do with your class:

Unplugged Activity Using Paper Blocks

Using the remote from the **The Big Event (Courses A, B) - Worksheet** and **Unplugged Blockly Blocks (Grades K-1) - Manipulatives**, gather your class to reprise the activity from the previous lesson. Ask the class "when the teal button is pushed, what do we do?" then fill in one of the `when` event blocks and one of the blue action blocks accordingly. Make sure that the students understand that the `when` blocks need to be on top of the blue block and they need to touch in order for the program to run.

💡 Lesson Tip

Students will have the opportunity to share their final product with a link. This is a great opportunity to show your school community the great things your students are doing. Collect all of the links and keep them on your class website for all to see!

Remind the students to only share their work with their close friends or family. For more information watch or show the class **Pause and Think Online - Video**.

Previewing Online Puzzles as a Class

Pull a puzzle from the corresponding online puzzles. We recommend puzzle 4 of this stage. Call on different students to make a funny face representing a mood when you click on Daisy. Explain this is an event that they are reacting to and Daisy can be coded to change moods when you click on her.

Main Activity (30 min)

Course B Online Puzzles - Website

This is the most free-form plugged activity of the course. At the final stage students have the freedom to create a story of their own. You may want to provide structured guidelines around what kind of story to write, particularly for students who are overwhelmed by too many options.

Wrap Up (5 - 10 min)

Journaling

Having students write about what they learned, why it's useful, and how they feel about it can help solidify any knowledge they obtained today and build a review sheet for them to look to in the future.

Journal Prompts:

- What was today's lesson about?
- Draw one of the **Feeling Faces - Emotion Images** that shows how you felt about today's lesson in the corner of your journal page.
- Draw an event you used in your program today.
- Imagine that you have a remote controlled robot. What would the remote look like? Draw a picture of what you think you could make the robot do.

Extended Learning

Use these activities to enhance student learning. They can be used as outside of class activities or other enrichment.

Look Under the Hood

When you share a link to your story, you also share all of the code that goes behind it. This is a great way for students to learn from each other.

- Post links to completed stories online
 - Make a story of your own to share as well!
- When students load up a link, have them click the "How it Works" button to see the code behind the story.
- Discuss as a group the different ways your classmates coded their stories.
 - What surprised you?
 - What would you like to try?
- Choose someone else's story and click Remix to build on it. (Don't worry, the original story will be safe.)

Standards Alignment

CSTA K-12 Computer Science Standards

- ▶ AP - Algorithms & Programming



This curriculum is available under a Creative Commons License (CC BY-NC-SA 4.0).

If you are interested in licensing Code.org materials for commercial purposes, **contact us**.