

Course C

Course C was developed for students in and around the second grade. It uses a limited understanding of shapes and elementary math concepts.

Students will create programs with loops, events, and conditionals. They will translate their initials into binary, investigate different problem-solving techniques, and discuss how to respond to cyberbullying. By the end of the course, students will create interactive games that they can share. Each concept in Course C is taught from the beginning, graduating toward experiences that allow for growth and creativity to provide all students a rich and novel programming experience.

Lesson 1: Building a Foundation

Unplugged

Build a structure that can hold a textbook. You might feel frustrated- remember to be persistent!

Lesson 2: Programming in Maze

Programming | Algorithms | Maze | Sequencing

Learn about sequences and algorithms with Angry Birds.

Lesson 3: Debugging in Maze

Debugging | Bug | Maze

Find problems in puzzles and practice your debugging skills.

Lesson 4: Real-Life Algorithms: Paper Planes

Unplugged | Algorithms | facilitating discussion

Let's make paper airplanes using algorithms!

Lesson 5: Programming in Collector

Collector | Program | Programming

Write algorithms to help Laurel the Adventurer collect lots of gems!

Lesson 6: Programming in Artist

Artist | Sequencing

Create beautiful images by programming the Artist.

Lesson 7: Getting Loopy

Unplugged | Loops

In this lesson, we'll have a dance party using repeat loops!

Lesson 8: Loops with Rey and BB-8

Loops | Maze

Help BB-8 through mazes using loops!

Lesson 9: Loops in Artist

Loop | Artist

In this lesson, loops make it easy to make even cooler images with Artist!

Lesson 10: Loops in Harvester

Loops | Harvester

Let's use loops to help the harvester collect some veggies!

Lesson 11: Events Unplugged: The Big Event

Unplugged | Events

Play a fun game to learn about events.

Lesson 12: Build a Flappy Game

Flappy | Event

Build your own Flappy Bird game however you like, then share it with your friends!

Lesson 13: Events in Play Lab

Play Lab | Event

It's time to get creative and make a game in Play Lab!

Lesson 14: Common Sense Education: Screen Out the Mean

Common Sense Education | Cyberbullying | Unplugged

In this lesson, you'll learn what to do if something online makes you feel angry, sad, or scared.

Lesson 15: Binary Bracelets

Unplugged | Binary

Create your very own binary bracelet and learn how computers remember information!



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: Building a Foundation

Overview

In this lesson, students are asked to build a load-bearing structure using common materials. The structure will be tested on its ability to hold a textbook for more than ten seconds. The goal of this activity is to engage students in a difficult challenge to highlight strategies for showing persistence in the face of frustration. Most students will not get this right the first time, but if they continue trying and iterating, you can help them identify techniques to make struggle productive.

Purpose

While this lesson may not at first seem connected to Computer Science, it plays an essential role in preparing students to tackle some of the more difficult challenges that will come their way when approaching new CS content. This lesson teaches that failure is not the end of a journey, but rather a step towards success. The majority of students should feel frustrated at some point in this lesson, but it's important to emphasize that failure and frustration are common steps that lead to creativity and success.

Agenda

Warm Up (15 min)

Try, Try Again

Activity (20 min)

**Building a Foundation
Testing and Iteration**

Wrap Up (10 min)

Vocabulary

Flash Chat: What did we learn?

[View on Code Studio](#)

Objectives

Students will be able to:

- Identify the feeling of frustration when felt or described
- List strategies for overcoming frustration during a difficult task
- Model persistence while working on a difficult task

Preparation

☐ Watch the Building a Foundation - Lesson in Action Video.

☐ Print Building a Foundation - Teacher Prep Guide.

☐ Gather enough building elements (marshmallows or gumdrops with toothpicks or popsicle sticks) for each group. You don't have to give any certain amount; just make sure you put some limit on materials.

☐ Give a Think Spot Journal to each student.

Links

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

For the Teachers

- **Building a Foundation** - Lesson in Action Video
- **Building a Foundation** - Teacher Prep Guide

For the Students

- **Think Spot Journal** - Reflection Journal

[Make a Copy](#)

Vocabulary

- **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 (15 min)

Try, Try Again

Think: Ask students to close their eyes and think of a time where they tried to do something and didn't succeed. Maybe it was attempting a new sport, playing a tough video game, or learning a new skill. As they are thinking silently, ask that they focus specifically on:

- What were you feeling?
- What were you thinking?
- What were you saying?
- What were you doing?

Pair: Without discussing **what** the actual situation was, have students share with a neighbor about what it looked to try something and not succeed. Encourage them to talk about the things they were feeling, thinking, saying, and doing.

Share: Once all of the students have had a chance to share thoughts with their neighbors, ask a few to share what they were talking about with the whole class. Keep track of common answers somewhere that students can refer to later in the lesson (and possibly during future lessons as well).

Discuss: Using the list that the class generated, discuss which responses to failure were productive (potentially led towards later success) and which were not. Highlight the constructive responses so that students can refer to them later.

Share: Share out a personal experience where you (the teacher) initially failed at something, but persisted through to success. Let students know that today we are all going to work on a challenge that is meant to be hard and even frustrating. Let them know that it was chosen because it's likely to create several failed attempts before it works but we can rely on some of the constructive responses we talked about earlier to move towards success.

Lesson Tip

As you work to embed persistence into your classroom culture, consider using some of these images as occasional reminders.

- **Mouse Wants a Cracker**
- **Fall 7 Times, Stand Up 8**
- **Never Ever Give Up**
- **If You Quit Too Soon**

Discussion

Focus on the experience: The goal of this discussion is to remind students what it feels like to fail, because we want to be able to identify those feelings when they occur so that we know when to rely on persistence strategies. Students may be tempted to share information about the specific memories they thought of, but redirect students to focus on their experience. Some expected answers include:

- Felt sad / mad
- Cried
- Yelled (or even cursed)
- Threw or broke things
- Asked for help
- Quit

Activity (20 min)

Building a Foundation

The goal of this lesson is not just building a structure, like the activity makes it appear. Instead, it's to prepare students to face failure and frustration with persistence. In order for that to happen, students need to know that they are not alone in feeling bad when things go wrong. They also need to be prepared to struggle. The outcome of this activity is much stronger when students know what they are about to experience.

Remarks

We are going to work in groups today to solve a challenge. It is going to be a very hard challenge, and they made it this way on purpose! Part of completing this challenge is knowing that we will struggle and we will probably even feel frustrated. Just know that if you hang in there through the failure, you will eventually succeed!

Set Up: You'll need to have a collection of supplies for each group ready for this activity. It's important that the available supplies are limited in number and structurally unsound. Potential supply kits include:

- 20 gumdrops and 50 toothpicks
- 20 drinking straws and 6" of tape
- 20 marshmallows and 20 craft sticks
- 10 playing cards and 4" of tape

Transition: Introduce the challenge, which is to build a structure that can hold a textbook for at least 10 seconds, using provided materials.

💡 **Group:** Divide students into groups of three or four.

Distribute: Pass out the building supplies, letting students know that they are limited to only the supplies you provide.

Display: Show rules that are similar to the ones itemized below in a place where students can easily see and refer back to as they work.

- Use only the supplies provided to build a structure.
- The structure can be any shape, but it has to be at least 2.5" tall. (Example)
- The structure must support the weight of a book for a full 10 seconds. (Example)

Prompt: Ask groups to spend at least 5 minutes planning a method for building their first tower. Each group should draw or write down the steps in their plan.

Circulate: Encourage students to begin building, and observe as they work. Keep your eyes open for signs of frustration and repeated failure. Make sure to acknowledge frustration and praise their persistence. Refer to the methods of dealing with frustration and the methods for productive persistence that are written on the board. Keep reminding individuals that the goal of this lesson is to experience frustration and persistence, so if they are failing, they're really succeeding!

When groups have reached a design they believe will meet conditions, move them to the "Testing and Iteration" phase.

Testing and Iteration

Model: The first time a group tests their structure, model for them how you intend to test in line with the rules. Specifically you'll want to:

- Check that it's at least 2.5" tall
- Ensure that they are using only provided materials
- Gently place a text book on top and time how long it holds out.

Prompt: If the group's structure failed, ask them to discuss:

- Why it failed
- Whether their plan needs revision, or just another try
- How they're going to improve the next iteration

If the structure succeeded, push them to support more weight until it breaks, and then follow the same reflection and iteration process.

💡 Teaching Tip

Providing Examples: The planning stage can be difficult for young students. It may be helpful for you to place some idea "examples" at the front of the room. Do not announce that they are there. Simply encourage students to take a walk if they get frustrated. Try to encourage students to locate the tips on their own if at all possible. This helps students feel like they "discovered" something that helped them, rather than being rescued. Make sure you also have "tips" for being persistent and for dealing with frustration written in a clear location.

Tips for Overcoming 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
- What is happening?
- What is supposed to happen?
- What does that tell you?
- Make a change and try again

Circulate: Allow as much time as you can for groups to continue iterating on their plans. Make sure to be excited not only for the successes, but also the failures. Model for students how much you can learn from your failed attempts.

Wrap Up (10 min)

Vocabulary

Display: Present the vocab for this lesson, persistence, frustrated. Let the class know that they were showing persistence when they worked through failures in their structures.

Flash Chat: What did we learn?

🗣️ **Discuss:** Reflect on the activity as a class, using the following prompts as a start.

- Did anyone feel frustrated during this lesson?
 - Can you share what that felt like?
- (Alternatively, you can ask "Who felt frustrated at some point during this lesson?" and cheer for the students who raise their hand.)
 - Then YOU won this lesson, and YOU won this lesson, and YOU won, too!
- Was there a time that you thought about giving up?
 - What were you feeling, saying, doing, or thinking at that time?
 - How did you get past that feeling?
- Do you think that you would be more proud of yourself for solving something that was easy, or something that was very, very hard?

Discussion

This Flash Chat is meant to help the students come to terms with the negative emotions that they felt during the project, and help them see that they were not alone in their struggle. It should also prepare them to recall the tools that they used to help themselves overcome frustration and be persistent so that they can access those tools again when solving online puzzles.

End the discussion by bringing the context back to the warm-up.

Think: Where would you be now if you had not been persistent when you were learning to walk as a baby? Or if you had been too frustrated to keep going when you were learning to talk? What other things did you learn to do, even though they were very, very hard?

Pair: Have students discuss this thought exercise with an elbow partner to come up with 1-3 things that fit that description.

Share: Have students write the answers in their journals to remind themselves how strong they can be when faced with a challenge.

Standards Alignment

CSTA K-12 Computer Science Standards (2017)

- 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 2: Programming in Maze

Overview

Using characters from the game Angry Birds, students will develop sequential algorithms to move a bird from one side of a maze to the pig 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 (4 min)

Review Unplugged Activity

Bridging Activity - Programming (10 min)

Transitioning from Unplugged to Online

Previewing Online Puzzles as a Class (3 min)

Main Activity (30 min)

Course C Online Puzzles - Website

Wrap Up (5 - 10 min)

Journaling

Extended Learning

View on Code Studio

Objectives

Students will be able to:

- Translate movements into a series of commands.
- Identify and locate bugs in a program.

Preparation

☐ Play through the **Course C Online Puzzles - Website** in stage 2 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 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 Teachers

- **Course C Online Puzzles - Website**
- **CS Fundamentals Main Activity Tips - Lesson Recommendations** [Make a Copy](#)

For the Students

- **Course C, Lesson 2 Maze Bridging Page** - Puzzle Manipulative (PDF)
- **Unplugged Maze Blocks** - Manipulatives
- **Think Spot Journal** - Reflection Journal

[Make a Copy](#)

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.
- **Sequencing** - Putting commands in correct order so computers can read the commands.

Teaching Guide

Warm Up (4 min)

Review Unplugged Activity

This lesson relies on many of the ideas that students learned while struggling with the "Building a Foundation" activity. It is important that you bring those concepts (such as persistence and debugging) around full-circle so that your class can benefit from them in their online work as well.

Display: Show students pictures of the "Building a Foundation" activity that you completed in the lessons prior to this one.

Discuss: Ask students why you put them through the challenging experience of creating a structure with hard rules.

- Do you remember any of the ideas that we learned?
- Persistence, Frustration, Debugging
- When you make a mistake, are you supposed to just give up? Do you throw everything away and start over?

Transition: Once you are satisfied that your students remember "Building a Foundation", you can move into the Bridging Activity.

Bridging Activity - Programming (10 min)

Transitioning from Unplugged to Online

This short activity will help students relate the ideas of persistence and debugging to the puzzles that they are about to complete online.

Display: Project a copy of the **Course C, Lesson 2 Maze Bridging Page - Puzzle Manipulative (PDF)** for the class to see. Make sure that you have pre-placed the movement blocks in the workspace using **Unplugged Maze Blocks - Manipulatives** in a configuration like the one below:



Model: Tell students that you have this workspace on display that looks just like the area that they will see when they start to do the Code.org puzzles online. As the teacher, let them know that you are SO SMART that you already put all of the code in that you are going to need to solve this puzzle, then ask them to watch you "Run" it by moving your finger (or a penny, or some other indicator) along the path.

It won't be long before you run into a block of TNT. Feign frustration.

Discuss: - What am I feeling right now, do you think? - Should I quit? - Should I throw all of the code away and start over?

Think: How can I fix this program so that I don't run into the TNT?

Pair: Have students work on solutions to get the bird around the TNT. Depending on your classroom, you might want to either have them fix each mistake one at a time (with demos in between) or students might feel comfortable working together to fix the entire program.

Share: Have volunteers come up to help move the blocks into the right location. "Run" the program over and over as a class, fixing bugs, until the bird does what it is supposed to. Continue to point out experiences that relate to persistence, frustration, and debugging.

When your class reaches the pig, celebrate not only their achievements, but their persistence!

Previewing Online Puzzles as a Class (3 min)

Students should now be ready to see a real puzzle in action!

Model: Pull up Lesson 2, Puzzle 5 to do in front of the class. This will be the same puzzle that they just saw in the bridging activity. While working through this puzzle with the class, remind students that making mistakes is okay and remind them that the only way to be successful is to be persistent. Tie the limitations of the online puzzles into the limitations that students were facing with "Building a Foundation". (When completing "Building a Foundation", you only had x toothpicks and y gumdrops. Here, you only have 3 move forward blocks and 2 turn blocks.) Remind them that finding the solution might be harder with limitations, but that you believe in them.

💡 Lesson Tip

Some students may struggle with turning their bird in the correct direction, particularly when the bird isn't facing up. Remind students that when we say turn left or right, we're giving directions from the bird's point of view.

Discuss: Does anyone remember how to solve this puzzle?

As the teacher, you should decide if you will have the students remind you how to solve it from their seats, or come to the computer to drag the actual blocks in one-by-one.

Transition: Now that students have seen an online puzzle in practice, they should be ready to start solving puzzles of their own. Continue to the lab or bring out their classroom machines.

Main Activity (30 min)

Course C 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 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.

Discuss: After providing students with end-of-class warnings, grab everyone's attention and get them to reflect on the experiences that they just had.

- Did anyone feel frustrated during any of the puzzles?
- Did anyone notice the need to be persistent?

Transition: Have students grab their Thinkspot Journals and take a moment to leave lessons for themselves.

💡 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

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?
- How did you feel during today's lesson?
- Draw an activity you like to do that you struggled with the first time. Draw or describe how you got better.

Extended Learning

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

Create Your Own

In small groups, let students design their own mazes and challenge each other to write programs to solve them. For added fun, make life-size mazes with students as the pig and bird.

Standards Alignment

CSTA K-12 Computer Science Standards (2017)

- ▶ **AP** - Algorithms & Programming



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

Lesson 3: Debugging in Maze

Overview

Debugging is an essential element of learning to program. In this lesson, students will encounter puzzles that have been solved incorrectly. They will need to step through the existing code to identify errors, including incorrect loops, missing blocks, extra blocks, and blocks that are out of order.

Purpose

Students in your class might become frustrated with this lesson because of the essence of debugging. **Debugging** is a concept that is very important to computer programming. Computer scientists have to get really good at facing the bugs in their own programs. Debugging forces the students to recognize problems and overcome them while building critical thinking and problem solving skills.

Agenda

Warm Up (15 min)

Introduction
Vocabulary

Main Activity (30 min)

Course C Online Puzzles - Website

Wrap Up (5 - 10 min)

Journaling

Extended Learning

[View on Code Studio](#)

Objectives

Students will be able to:

- Predict where a program will fail.
- Modify an existing program to solve errors.
- Reflect on the debugging process in an age-appropriate way.

Preparation

- ☐ Play through the **Course C Online Puzzles - Website** in stage 3 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**.
- ☐ Review **Debugging Recipe - Student Handout** with the class.

Links

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

For the Teachers

- **CS Fundamentals Main Activity Tips - Lesson Recommendations** [Make a Copy](#)

For the Students

- **Debugging Recipe - Student Handout** [Make a Copy](#)
- **Think Spot Journal - Reflection Journal** [Make a Copy](#)

Vocabulary

- **Bug** - Part of a program that does not work correctly.
- **Debugging** - Finding and fixing problems in an algorithm or program.

Teaching Guide

Warm Up (15 min)

Introduction

Ask students to think about problems they have to solve in everyday life.

- How do you fix something that isn't working?
- Do you follow a specific series of steps?
- The puzzles in this unit have already been solved for you (yay!), but they don't seem to be working (boo!)
- We call the problems in these programs "bugs," and it will be your job to "debug" them.

Vocabulary

This lesson has three new and important vocabulary words:

- **Bug** - Say it with me - Buhh-g. Something that is going wrong. An error.
- **Debugging** - Say it with me: Dee-bug-ing. To find and fix errors.
- **Persistence** - Say it with me: Purr-siss-tense. Not giving up. Persistence works best when you try things many different ways, many different times.

Say:

Debugging is a process. First, you must recognize that there is an error in your program. You then work through the program step by step to find the error. Try the first step, did it work? Then the second, how about now? If you make sure that everything is working line by line, then when you get to the place that your code isn't doing what it's supposed to, you know that you've found a bug. Once you've discovered your bug, you can work to fix (or "debug") it!

If you think it will build excitement in the class you can introduce the character of today's puzzles, Scrat from Ice Age. If students aren't familiar with Scrat, **show some videos** of the quirky squirrel running into trouble.

Main Activity (30 min)

Course C Online Puzzles - Website

Before letting the students start on the computer, remind them of the advantages of **Pair Programming - Student Video** and asking their peers for help. Sit students in pairs and recommend they ask at least two peers for help before they come to a teacher.

As mentioned in the purpose of this lesson, make sure the students are aware that they will face frustrating puzzles. Tell them it is okay to feel frustrated, but it is important to work through the problem and ask for help. As the students work through the puzzles, walk around to make sure no student is feeling so stuck that they aren't willing to continue anymore.

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?
- How did you feel during today's lesson?
- What kind of bugs did you find today?

- Draw a bug you encountered in one of the puzzles today. What did you do to "debug" the program?

Extended Learning

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

Planting bugs

Have students go back through previous levels, purposefully adding bugs to their solutions. They can then ask other students to debug their work. This can also be done with paper puzzles.

When other students are debugging, make sure that the criticisms are constructive. If this could be a problem for your class, go over respectful debugging before this activity by role playing with another student.

Standards Alignment

CSTA K-12 Computer Science Standards (2017)

- ▶ **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 4: Real-Life Algorithms: Paper Planes

Overview

In this lesson, students will relate the concept of algorithms back to everyday activities. After discussing their steps, students will make paper planes using an algorithm. The goal here is to start building the skills to translate real world situations to online scenarios and vice versa.

Purpose

This lesson exists to help students see that an "algorithm" is just a list of steps that someone can take to finish a task. Students will also learn that the order of the individual steps can make a difference in the final product. This should help lay a foundation of understanding for arranging blocks into programs.

Agenda

Warm Up (15 min)

Main Activity (20 min)

Wrap Up (15 min)

Journaling

Assessment (15 min)

Extended Learning

View on Code Studio

Objectives

Students will be able to:

- Decompose large activities into a series of smaller events.
- Organize sequential events into their logical order.

Preparation

☐ Watch the Real-Life Algorithms: Paper Planes - Lesson in Action Video.

☐ Gather paper for students to construct paper airplanes from.

☐ Print out Real-Life Algorithms: Paper Planes - Worksheet for each student.

☐ Print Daily Algorithms - Assessment for each student.

☐ Make sure every student has a Think Spot Journal.

Links

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

For the Teachers

- **Real-Life Algorithms: Paper Planes** - Unplugged Video (**download**)
- **Real-Life Algorithms: Paper Planes** - Lesson in Action Video
- **Real-Life Algorithms: Paper Planes** - Worksheet Answer Key **Make a Copy**

For the Students

- **Real-Life Algorithms: Paper Planes** - Worksheet **Make a Copy**
- **Daily Algorithms** - Assessment **Make a Copy**
- **Think Spot Journal** - Reflection Journal **Make a Copy**

Vocabulary

- **Algorithm** - A list of steps to finish a task.

Teaching Guide

Warm Up (15 min)

This warm-up will help your students get perspective on what it means to create a list of steps to finish a task.

Prompt: Ask your students what they did to get ready for school this morning.

Discuss: Write their answers on the board. If possible, put numbers next to their responses to indicate the order that they happen. If students give responses out of order, have them help you put them in some kind of logical order, pointing out places where order matters and places where it doesn't.

Say: Introduce students to the idea that it is possible to create algorithms for the things that we do everyday. Give them a couple of examples, such as making breakfast, brushing teeth, and planting a flower.

Main Activity (20 min)

Real-Life Algorithms: Paper Planes - Worksheet

You can use algorithms to help describe things that people do every day. In this activity, we will create an algorithm to help students fold a paper plane.

Pair: Have students pair up with an elbow partner to cut the individual steps apart. They should discuss among themselves which six are needed and which three can be recycled.

Distribute: Hand out some blank paper (or the back of other papers) for students to glue/tape their instructions to once they've decided on the order. Make sure they have a second piece that they can actually fold into the final plane.

Share: Finally, have students trade their finished algorithms with another pair and have them use one another's algorithms to make a paper plane.

Circulate: Walk around during the activity and check that students are trying to debug mistakes. If they get stuck, have them check-in with the group who originally coded their algorithm and see if they can work together to fix it.

Discuss: What did we learn?

- How many of you were able to follow your classmates' algorithms to make your planes?
- Did we leave anything out when making the plane?
 - What would you have added to make the algorithm even better?
 - What if the algorithm had been only one step: "Fold a Paper Plane"?
 - Would it have been easier or harder?
 - What if it were forty steps?
- What was your favorite part about this activity?

💡 Lesson Tip

If you think it will be too difficult for your students to decide which steps are correct, do that piece together as a class before you break up into pairs.

💡 Lesson Tip

If you are concerned about injury when your students begin flying their paper planes, we recommend having them blunt the tip of the plane by either folding it inward or ripping it off and covering the ripped edges with tape.

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?
- How did you feel during today's lesson?
- Can you imagine an algorithm for building a real plane? What do you think that would look like?
- Write out an algorithm that would take you from your desk to the front of the class.

Assessment (15 min)

Daily Algorithms - Assessment

- Hand out the **Daily Algorithms - Assessment** 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.

Go Figure

- Break the class up into teams.
- Have each team come up with several steps that they can think of to complete a task.
- Gather teams back together into one big group and have one team share their steps, without letting anyone know what the activity was that they had chosen.
- Allow the rest of the class to try to guess what activity the algorithm is for.

Standards Alignment

CSTA K-12 Computer Science Standards (2017)

- ▶ **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 in Collector

Overview

In this series of puzzles, students will continue to develop their understanding of algorithms and debugging. With a new character, Laurel the Adventurer, students will create sequential algorithms to get Laurel to pick up treasure as she walks along a path.

Purpose

In this lesson, students will be practicing their programming skills using a new character, Laurel the Adventurer. When someone starts **programming** they piece together instructions in a specific order using something that a machine can read. Through the use of programming, students will develop an understanding of how a computer navigates instructions and order. Using a new character with a different puzzle objective will help students widen their scope of experience with sequencing and algorithms in programming.

Agenda

Warm Up (5 min)

Introduction

Bridging Activity - Programming (10 min)

Previewing Online Puzzles as a Class

Main Activity (30 min)

Course C Online Puzzles - Website

Wrap Up (5 - 10 min)

Journaling

[View on Code Studio](#)

Objectives

Students will be able to:

- Order movement commands as sequential steps in a program.
- Represent an algorithm as a computer program.
- Develop problem solving and critical thinking skills by reviewing debugging practices.

Preparation

☐ Play through **Course C Online Puzzles - Website** stage 5 to find 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 Teachers

- **Course C Online Puzzles** - Website
- **CS Fundamentals Main Activity Tips** - Lesson Recommendations [Make a Copy](#)

For the Students

- **Unplugged Blocks (Courses C-F)** - Manipulatives [Make a Copy](#)
- **Think Spot Journal** - Reflection Journal [Make a Copy](#)

Vocabulary

- **Algorithm** - A list of steps to finish a task.
- **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 (5 min)

Introduction

This lesson uses most of the same blocks from Course C, Stage 2 Programming in Maze and adds the ability to collect .

Tell the students that this block will allow Laurel the Adventurer to pick up the treasure that she is standing over. This new block will be discussed more in the bridging activity.

Bridging Activity - Programming (10 min)

This activity will help bring the unplugged concepts from **Real-Life Algorithms: Paper Airplanes** into the online world that the students are moving into.

Previewing Online Puzzles as a Class

Pull a puzzle from the corresponding online stage. We recommend puzzle 7. Have students discuss a pattern that they think will get Laurel the Adventurer to collect all the treasure. Ask the students to share. See how many other students had the same answer!

Main Activity (30 min)

Course C Online Puzzles - Website

Laurel the Adventurer is looking to collect as much treasure as she can. Instruct the students to traverse the puzzle to collect whatever they can. Some levels will require you to only pick up one piece of treasure, but others will require you to pick up every piece of treasure. Pay attention to the instructions to know what to do!

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?
- How did you feel during today's lesson?
- Draw a maze that you might solve with the blocks you used today.

Standards Alignment

CSTA K-12 Computer Science Standards (2017)

- 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: Programming in Artist

Overview

In this lesson, students will take control of the Artist to complete drawings on the screen. This Artist stage will allow students to create images of increasing complexity using new blocks like `move forward by 100 pixels` and `turn right by 90 degrees`.

Purpose

Building off of the students' previous experience with sequencing, this lesson will work to inspire more creativity with coding. The purpose of this lesson is to solidify knowledge on sequencing by introducing new blocks and goals. In this case, students learn more about pixels and angles using the new blocks, while still practicing their sequencing skills. Also, students will be able to visualize new goals such as coding the Artist to draw a square.

Agenda

Warm Up (10 min)

Introduction

Main Activity (30 min)

Course C Online Puzzles - Website

Wrap Up (10 - 15 min)

Journaling

Extended Learning

View on Code Studio

Objectives

Students will be able to:

- Create a program to complete an image using sequential steps.
- Break complex shapes into simple parts.

Preparation

☐ Play through the **Course C Online Puzzles - Website** in stage 6 to find any potential problem areas for your class.

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

☐ (Optional) Obtain protractors for your class to visualize the angles they must use to complete the puzzles.

☐ Print one **Turns & Angles - Student Handout** for each student.

Links

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

For the Teachers

- **Course C Online Puzzles - Website**
- **CS Fundamentals Main Activity Tips - Lesson Recommendations** [Make a Copy](#)

For the Students

- **Artist Introduction - Student Video**
- **Turns & Angles - Student Video**
- **Turns & Angles - Student Handout** [Make a Copy](#)
- **Think Spot Journal - Reflection Journal** [Make a Copy](#)

Teaching Guide

Warm Up (10 min)

Introduction

Show the students one or both of the following videos as an introduction to angles:

Artist Introduction - Student Video (1.5 minutes long)

Turns & Angles - Student Video (2 minutes long)

Use the **Turns & Angles - Student Handout** to show the students interior versus exterior angles for different shapes. This document can be used as a hand out or you can choose to print it out as a poster for students to refer to.

Ask:

Discuss the square and triangle shapes from the document.

- How would you code a computer to draw that shape?
- What order do the instructions need to be in?

Tell the students that in these puzzles they will be moving a character who leaves a line everywhere he goes. The students will be writing code that gets the character to draw various shapes, including a square.

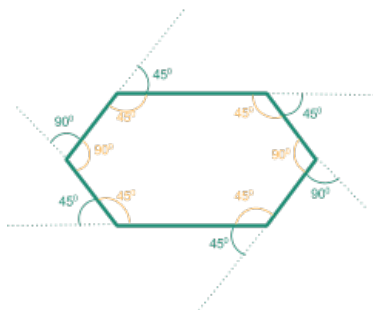
Main Activity (30 min)

Course C Online Puzzles - Website

In this set of puzzles, the artist will no longer be constrained to 90 degree angles. Having physical protractors available can help students better visualize the angles they need. Otherwise, the stage provides images of the angles as the student selects which angle to use. (Please note: Angle choices are limited to two inside of the dropdown menu, reducing the number of options students have to work through.)

Before sending the students to the computers to work on the puzzles, it might be beneficial to give a brief presentation of how to use the tools in this level. We recommend puzzle 5 as a good puzzle to show how to use the protractor online.

The eighth puzzle asks the students to draw a 6 sided polygon. This might be challenging for some students. We recommend getting the students to try a few times, ask a peer, then ask the teacher for help. Below is an image that might be helpful for the students.



Wrap Up (10 - 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?
- How did you feel during today's lesson?
- What are the interior angles that make up a square. What about for a triangle?
- Sketch a simple shape on your paper and imagine the code used to draw it. Can you write that code out next to the shape?

Extended Learning

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

The Copy Machine

- Give students two pieces of paper
- On one sheet draw a simple image, using straight lines only.
- On the second sheet draw instructions for recreating that image commands to move straight and turn at various angles.
- Trade instruction sheets and attempt to recreate the image using only the provided instructions.

Standards Alignment

CSTA K-12 Computer Science Standards (2017)

- ▶ **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: Getting Loopy

Overview

As we start to write longer and more interesting programs, our code often contains a lot of repetition. In this lesson, students will learn about how loops can be used to more easily communicate instructions that have a lot of repetition by looking at the repeated patterns of movement in a dance.

Purpose

At this point in the course, students should have developed comfort with programming a set of linear instructions. Frequently the linear set of instructions includes patterns that are repeated multiple times and as students want to write more complex and interesting programs, manually duplicating that code becomes cumbersome and inefficient. To enable students to write more powerful programs, we'll need to rely on structures that break out of the that single linear list. **Loops** allow for students to structure their code in a way that repeats. In this lesson, we will focus on identifying patterns in physical movement before moving back onto the computer to look for patterns in our code.

Agenda

Warm Up (5 min)

Repeat After Me

Main Activity (15 min)

Assessment (10 min)

Wrap-Up (15 min)

Vocabulary

Flash Chat: What did we learn?

Journaling

Extended Learning

So Moving

Connect It Back

[View on Code Studio](#)

Objectives

Students will be able to:

- Repeat actions initiated by the instructor.
- Translate a picture program into a real-world dance.
- Convert a series of multiple actions into a single loop.

Preparation

- ☐ Print one Getting Loopy - Worksheet for the class.
- ☐ Print one Getting Loopy - Assessment for each student.
- ☐ Make sure every student has a Think Spot Journal.

Links

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

For the Teachers

- **Getting Loopy** - Assessment Answer Key
[Make a Copy](#)

For the Students

- **Getting Loopy** - Unplugged Video ([download](#))
- **Getting Loopy** - Worksheet
[Make a Copy](#)
- **Getting Loopy** - Assessment
[Make a Copy](#)
- **Think Spot Journal** - Reflection Journal
[Make a Copy](#)

Vocabulary

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

Teaching Guide

Warm Up (5 min)

Repeat After Me

Model: Ask for a volunteer and have them stand.

- Instruct your volunteer to walk around the table (or their chair, or a friend).
- When they finish, instruct them to do it again, using the exact same words you did before.
- When they finish, instruct again.
- Then again.

Prompt: Would it have been easier for me to just ask you to go around the table four times?

Think: What if I wanted you to do it ten times? How would you reword my instructions so that they were more efficient and I didn't have to repeat myself so much? Feel free to write your instructions down on a piece of scrap paper.

Share: Ask a few students to share their instructions with the class, pointing out how each approach has simplified the overall approach to giving instructions.

Remarks

Today we're going to work on finding ways to make giving lots of instructions easier, especially when those instructions repeat themselves a lot. This will be really useful when we go back to the computers and have to write lots of instructions in our programs.

Main Activity (15 min)



Dance Party

Say: Introduce the main activity by letting the class know that we will be having a dance party. In order to have that party, we'll need to know what all of the steps in the dance are, and how many times we should do them.

Display: Show the **Getting Loopy - Worksheet** so that all students can see it. Talk through the different sections of the dance as a class. Point out the section that repeats, in particular.

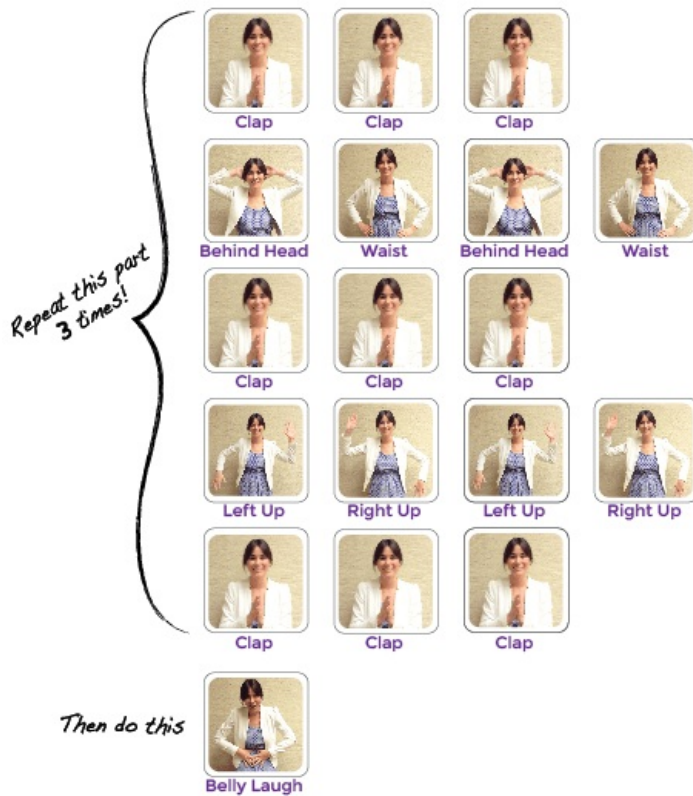
Lesson Tip:

Looking for some good music? Here are some great places to find some:

- **Radio Disney**
- **Nick Radio**
- **Kidz Bop Radio**

Please be advised that some of these stations may display ads with third-party content. If you find that displayed ads are inappropriate, you may want to direct students to a different site, or research ad-blockers that can prevent this content.

The Iteration



Model: Show the class what the entire dance looks like done at full-speed. Then run through the dance slowly, asking a different student to call out each line of instructions. Next, have the students perform the dance along with you, saying the instructions aloud as they get to each move.

Prompt: Ask students to work with a neighbor to find all of the sections of the dance that repeat.

Share: Ask a few students to share the repeating patterns that they found. As a class, talk through how you might rework the instructions to be even shorter by repeating those patterns.

Finally, help them understand a symbology for capturing these loops on their picture program, since the assessment will utilize this same method. Here is an example:



Assessment (10 min)

Ending with an assessment sheet will help solidify this lesson for your students.

Distribute: Hand out the **Getting Loopy - Assessment** to each student. Allow students to complete the activity independently after the instructions have been well explained. This should feel familiar, thanks to the previous activities.

Wrap-Up (15 min)

Vocabulary

Display: Present the vocab for this lesson, loop. Ask the class to point out the main loop that was in the dance. Why do you think we call it a loop?

Flash Chat: What did we learn?

- Do you think it is easier to add more pictures to the screen or change the number of times we loop?
 - Would your answer be the same if we wanted to loop 100 times?
- Could we use these same loops with different dance moves?
- Do you know any dances that are done inside a loop?
- What was your favorite part about that activity?

Journaling

Having students write or draw 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?
- How did you feel during today's lesson?
- Draw a picture of you dancing today. Draw the loops that you did, like clapping three times.
- What else can you use a loop for?

Extended Learning

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

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 (2017)

- ▶ **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: Loops with Rey and BB-8

Overview

Building on the concept of repeating instructions from "Getting Loopy," this stage will have students using loops to help BB-8 traverse a maze more efficiently than before.

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 the Code.org puzzles, students will learn to add instructions to existing loops, gather repeated code into loops, and recognize patterns that need to be looped. It should be noted that students will face puzzles with many different solutions. This will open up discussions on the various ways to solve puzzles with advantages and disadvantages to each approach.

Agenda

Warm Up (5 min)

Introduction

Bridging Activity - Loops (10 min)

Unplugged Activity Using Paper Blocks Previewing Online Puzzles as a Class

Main Activity (30 min)

Course C Online Puzzles - Website

Wrap Up (5 - 10 min)

Journaling

Extended Learning

[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 largest repeatable sequence.
- Employ a combination of sequential and looped commands to reach the end of a maze.

Preparation

☐ Play through the **Course C Online Puzzles - Website** in stage 8 to determine if there will be any 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 Teachers

- **Course C Online Puzzles** - Website
- **CS Fundamentals Main Activity Tips - Lesson Recommendations** [Make a Copy](#)

For the Students

- **Unplugged Blocks (Courses C-F) - Manipulatives** [Make a Copy](#)
- **Think Spot Journal - Reflection Journal** [Make a Copy](#)

Vocabulary

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

Teaching Guide

Warm Up (5 min)

Introduction

Review with students the "Getting Loopy" activity:

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

Quickly show the students a dance (with repeated steps) for the rest of the class to do. Ask the rest of the class to find the loops within the dance and point them out.

If you're comfortable, give an introduction to BB-8 from Star Wars. Many children may be familiar with the lovable robot, but an introduction will surely build excitement.

Bridging Activity - Loops (10 min)

This activity will help bring the unplugged concepts from "Getting Loopy" 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

Revisit the dance from "Getting Loopy." This time, work with the class to "code" it out using **Unplugged Blocks (Courses C-F) - Manipulatives** instead of writing the dance instructions on paper. Make sure the students know that the blocks need to go from top to bottom and they all need to touch!

Previewing Online Puzzles as a Class

Pull up the online puzzles and choose a puzzle to do in front of the class. We recommend puzzle 8 for its staircase pattern. Ask the students to write a program to solve the puzzle on paper. Have the students circle repeated chunks and label with the number of repeats, the same way they did in "Getting Loopy."

Main Activity (30 min)

Course C Online Puzzles - Website

As students work through the puzzles, see if they can figure out how many blocks they use with a loop vs. not using a loop.

Pair Programming - Student Video works really well with this set of puzzles because there are a few ways to fill the loops. Push for friendly discussion between pairs in instances of disagreement on how to solve the puzzle. Have the students ask each other questions like:

- How did you come up with that solution?
- What are some benefits of solving the puzzle that way?

We also recommend having paper on hand for students to write out their code and find any repetition to use in loops.

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?
- How did you feel during today's lesson?
- How did loops make your program easier to write?
- Think of something that repeats over and over again. What might the program for that look like?

Extended Learning

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

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 (2017)

- **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 in Artist

Overview

Watch student faces light up as they make their own gorgeous designs using a small number of blocks and digital stickers! This lesson builds on the understanding of loops from previous lessons and gives students a chance to be truly creative. This activity is fantastic for producing artifacts for portfolios or parent/teacher conferences.

Purpose

This series highlights the power of loops with creative and personal designs.

Offered as a project-backed sequence, this progression will allow students to build on top of their own work and create amazing artifacts.

Agenda

Warm Up (15 min)

Introduction

Main Activity (30 min)

Course C Online Puzzles - Website

Wrap Up (15 min)

Journaling

View on Code Studio

Objectives

Students will be able to:

- Identify the benefits of using a loop structure instead of manual repetition.
- Differentiate between commands that need to be repeated in loops and commands that should be used on their own.

Preparation

- ☐ Play through the **Course C Online Puzzles - Website** corresponding to this course 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 Teachers

- **Course C Online Puzzles** - Website
- **CS Fundamentals Main Activity Tips** - Lesson Recommendations [Make a Copy](#)

For the Students

- **Think Spot Journal** - Reflection Journal [Make a Copy](#)

Vocabulary

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

Teaching Guide

Warm Up (15 min)

Introduction

Students should have had plenty of introduction to loops at this point. Based on what you think your class could benefit from, we recommend:

- Creating a new dance with loops just like in "Getting Loopy"
- As a class, playing through a puzzle from the last lesson, "Loops with Rey and BB-8"
- Reviewing how to use Artist by playing through a puzzle from "Programming in Artist"
- Previewing a puzzle from this lesson

All of these options will either review loops or the artist, which will help prepare your class for fun with the online puzzles!

Main Activity (30 min)

Course C Online Puzzles - Website

Some students may discover where to add repeat loops by writing out the program without loops then circling sections of repetitions. If the students in your class seem like they could benefit from this, have them keep paper and pencils beside them at their machines. Students might also enjoy drawing some of the shapes and figures on paper before they program it online. (When drawing stamps, it can be easier to symbolize those with simple shapes like circles and squares.)

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?
- How did you feel during today's lesson?
- What was the coolest shape or figure you programmed today? Draw it out!
- What is another shape or figure you would like to program? Can you come up with the code to create it?

Standards Alignment

CSTA K-12 Computer Science Standards (2017)

- **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 in Harvester

Overview

In the preceding stage, students used loops to create fantastic drawings. Now they're going to loop new actions in order to help the harvester collect multiple veggies growing in large bunches.

Purpose

It may seem unnecessarily repetitive to have two plugged stages introducing loops, but the practice of using loops for different reasons develops a student's understanding of what loops can do. In "Loops in Maze" students only used loops to repeat movements. In this lesson, students will use loops to repeat other actions like harvesting pumpkins. New patterns will emerge and students will use creativity and logical thinking to determine what code needs to be repeated and how many times.

Agenda

Warm Up (5 - 10 min)

Introduction

Main Activity (30 min)

Wrap Up (10 min)

Journaling

Extended Activity

View on Code Studio

Objectives

Students will be able to:

- Write a program for a given task which loops a single command.
- Identify when a loop can be used to simplify a repetitive action.
- Employ a combination of sequential and looped commands to move and perform actions.

Preparation

- ☐ Play through the **Course C Online Puzzles - Website** 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 Teachers

- **Course C Online Puzzles** - Website
- **CS Fundamentals Main Activity Tips** - Lesson Recommendations **Make a Copy**

For the Students

- **Think Spot Journal** - Reflection Journal **Make a Copy**

Vocabulary

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

Teaching Guide

Warm Up (5 - 10 min)

Introduction

At this point, students have only used loops in the Maze puzzles. Those puzzles focused on looping movement instructions. This stage will use loops to pick up multiple items in the same spot.

Ask:

- What are some other elements of our programs that could benefit from loops?
- Harvester puzzles can have lots of items in one spot. How can we use loops to pick up any number of them with just two blocks?

Main Activity (30 min)

Course C Online Puzzles - Website

When students are using loops to repeat an action (such as harvesting pumpkins), encourage them to think about the movements before and after that action. Could those actions be brought into the loop as well?

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:

- What was today's lesson about?
- How did you feel during today's lesson?
- Give two examples of when you used loops in your code.
- What else could a farmer harvest? Draw the code block that you would need to harvest that item.

Extended Activity

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

Harvesting Supplies

- Let students create piles of school supplies at their desks (pencils, erasers, etc.)
- Have their partners figure out how a harvester would walk from pile to pile, collecting each group of items along the way.
- Share the programs with the rest of the class.

Standards Alignment

CSTA K-12 Computer Science Standards (2017)

- **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: Events Unplugged: The Big Event

Overview

Students will soon learn that events are a great way to add flexibility 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. Events can make your program more interesting and interactive.

Purpose

Today, students will learn to distinguish events and 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 (Course C) - Event Controller

Wrap Up (10 min)

Flash Chat: What did we learn?

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 movements of the teacher as signals to initiate commands.
- Practice differentiating pre-defined actions and event-driven ones.

Preparation

☐ Print one The Big Event (Courses C-F) - Worksheet and Event Controller.

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

☐ Make sure every student has a Think Spot Journal.

Links

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

For the Teachers

- **The Big Event** - Assessment Answer Key

[Make a Copy](#)

For the Students

- **The Big Event** - Unplugged Video ([download](#))
- **The Big Event (Course C)** - Event Controller [Make a Copy](#)
- **The Big Event** - Assessment [Make a Copy](#)
- **Think Spot Journal** - Reflection Journal [Make a Copy](#)

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 event is 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 practice changing programs by introducing events.

Main Activity (15 min)

The Big Event (Course C) - Event Controller

- Do you remember guiding Red from Angry Birds to the pig in the Maze puzzles?
 - In that exercise, you knew in advance exactly where you wanted Red to go, so you could make a program that took Red 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.

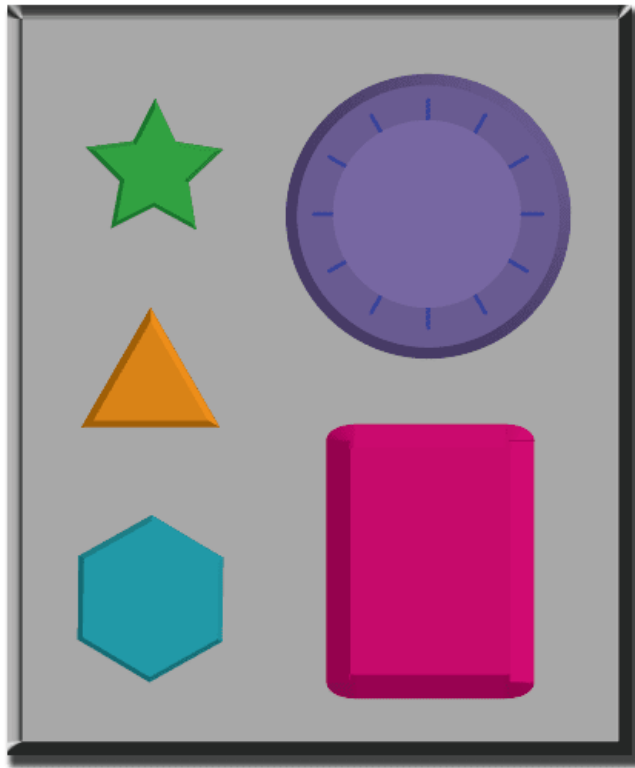
💡 Lesson Tip

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 other character 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!"
 - Green Button -> Clap
 - Orange Dial -> Stomp
- 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?

Assessment (10 min)

The Big Event - 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.

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 (2017)

- ▶ **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: Build a Flappy Game

Overview

In this special stage, students get to build their own Flappy Bird game by using event handlers to detect mouse clicks and object collisions. At the end of the level, students will be able to customize their game by changing the visuals or rules.

Purpose

Events are very common in computer programs. In this lesson, students will further develop their understanding of events by making a Flappy Bird game. Students will learn to make their character move across the screen, make noises, and react to obstacles based on user-initiated events.

Agenda

- Warm Up (10 min)
- Introduction
- Bridging Activity - Events (10 min)
- Unplugged Activity Using Paper Blocks
- Preview of Online Puzzles as a Class
- Main Activity (30 min)
- Course C Online Puzzles - Website
- Wrap Up (10 - 15 min)
- Journaling
- Extended Learning

View on Code Studio

Objectives

- Students will be able to:
- Match blocks with the appropriate event handler.
 - Create a game using event handlers.
 - Share a creative artifact with other students.

Preparation

- ☐ Play through **Course C Online Puzzles - Website** in stage 11 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 Teachers

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

Make a Copy

For the Students

- Think Spot Journal - Reflection Journal**

Make a Copy

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 click events to do?
- Now we're going to add events to our coding. Specifically, we're going to create an event for clicking the mouse and one for when the bird hits an object like the ground or an obstacle. When have you seen a character touch another object as an event in games?

💡 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. Monitor and collect all of the links to the projects and keep them on your class website for all to see!

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 (Course C) - Event Controller** and **Unplugged Blocks (Courses C-F) - 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.

Preview of Online Puzzles as a Class

Pull a lesson from the corresponding online stage, we recommend puzzle 2. Ask the students what should happen when the Flappy Bird runs into something like the ground or an obstacle. Explain that Flappy in this game will move forward with a click of the mouse and the game will end if Flappy runs into anything.

Complete the puzzle with the class and allow time for a quick discussion on what was and wasn't an event. For every event, ask the students what the action corresponding to this event is.

Main Activity (30 min)

Course C Online Puzzles - Website

In the final stage of this lesson students are able to tweak their game to make it unique - encourage them to see how different they can make each game within the constraints provided. If the class doesn't use **Pair Programming - Student Video**, then tell students to go around and look at other student's games. Otherwise, have students discuss and try out different ways to set up their game with their partner.

💡 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**.

Wrap Up (10 - 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?
- How did you feel about today's lesson?
- What did you do to make your game unique?
- Draw out a game you want to make in the future.

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 game, 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 games online or on the board.
 - Make a game 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 game.
- Discuss as a group the different ways your classmates coded their games.
 - What surprised you?
 - What would you like to try?
- Choose someone else's game and build on it. (Don't worry; the original game will be safe.)

Standards Alignment

CSTA K-12 Computer Science Standards (2017)

- **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: Events in Play Lab

Overview

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

Purpose

Here, students will further develop their understanding of events using Play Lab. Students will use events to make characters move around the screen, make noises, and change backgrounds based on user input. At the end of the puzzle sequence, students will be presented with the opportunity to share their projects.

Agenda

Warm Up (10 min)

Introduction

Main Activity (30 min)

Course C Online Puzzles - Website

Wrap Up (15 min)

Journaling

Extended Learning

View on Code Studio

Objectives

Students will be able to:

- Create an animated, interactive game using sequence and event-handlers.
- Identify actions that correlate to input events.

Preparation

- ☐ Play through the **Course C Online Puzzles - Website** in stage 12 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 Teachers

- **Course C Online Puzzles - Website**
- **CS Fundamentals Main Activity Tips - Lesson Recommendations** [Make a Copy](#)

For the Students

- **Unplugged Blocks (Courses C-F) - Manipulatives** [Make a Copy](#)
- **Think Spot Journal - Reflection Journal** [Make a Copy](#)

Vocabulary

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

Teaching Guide

Warm Up (10 min)

Introduction

Briefly discuss the Flappy Bird game from the last lesson. Ask students to come up with the various events in the game. Events include:

- Flappy hitting the ground
- Flappy hitting an obstacle
- A player clicking the screen
- Flappy passing an obstacle

Now discuss the actions that correspond to these events. Flappy running into something ends the game, but Flappy passing an obstacle wins a point. A player clicking makes Flappy flap his wings.

Ask the students what other events and actions they'd like to see. What other kinds of games could be built around those events and actions?

Main Activity (30 min)

Course C Online Puzzles - Website

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

💡 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**.

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?
- How did you feel about today's lesson?
- What is an event your program used today?
- Is there an event that would you like to have used in your game that you did not get to use in Play Lab?

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 (2017)

- ▶ **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 14: Common Sense Education: Screen Out the Mean

Overview

This lesson helps children to recognize that it is essential to tell a trusted adult if something online makes them feel angry, sad, or scared.

Students learn that other people can sometimes act like bullies when they are online. They will explore what cyberbullying means and what they can do when they encounter it. After reading a scenario about mean online behavior, students discuss what cyberbullying is, how it can make people feel, and how to respond. Finally, they use their knowledge to create a simple tip sheet on cyberbullying in their journal.

Purpose

Students may not ever have the misfortune of experiencing cyberbullying, but we want to make sure that the students are prepared for and knowledgeable about it, in case they ever witness it during an online situation. Students will learn how to identify cyberbullying and what steps they should take to make it stop. This may become helpful in later puzzles when students have the opportunity to share their work. If someone negatively responds to a student's work, this lesson will provide them with the tools that they need to handle the situation.

Agenda

Warm Up (5 min)

Introduction

Main Activity (35 min)

What Is Cyberbullying?

What to Do About Cyberbullying

Wrap Up (15 min)

Flash Chat: What did we learn?

Journaling

Assessment (5 - 10 min)

resource csf-screen-out-mean not found

[View on Code Studio](#)

Objectives

Students will be able to:

- Analyze online behaviors that could be considered cyberbullying.
- Explain how to deal with a cyberbullying situation.
- Recognize the importance of engaging a trusted adult if the student experienced cyberbullying.

Preparation

☐ Review **resource csf-screen-out-mean not found** from Common Sense Education's website.

☐ Print out a worksheet from the link above (page 6) for each student.

☐ Print out an assessment from the link at the top (page 7) for each student.

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

☐ Print or display the **Online Safety Poster - Student Handout** for the class to see.

☐ 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 Teachers

- **Common Sense Education** - Website
- **CSF Digital Citizenship** - Resource List

For the Students

- **Screen Out the Mean** - Worksheet

[Make a Copy](#)

- **Online Safety Poster** - Student Handout

[Make a Copy](#)

- **Think Spot Journal** - Reflection Journal

[Make a Copy](#) ▼

Vocabulary

- **Cyberbullying** - Doing something on the internet, usually again and again, to make another person feel angry, sad, or scared.
- **Online** - Connected to the internet.

Teaching Guide

Warm Up (5 min)

Introduction

Encourage:

Encourage students to share what they know about bullying.

Ask:

- What kinds of things count as bullying?
 - Students should understand that bullying is behavior that is purposefully mean or scary to someone else. For example, making fun of how someone looks, telling lies about them, or threatening to do something bad to them.
- How does bullying make other people feel?
 - Hurt, angry, upset, scared
- What is the best thing to do when you feel bullied, or when you see someone else being bullied?
 - Students should know to always tell a trusted adult when they experience or witness bullying.

Explain:

Students will be learning about a kind of bullying that can take place when they use the internet.

Main Activity (35 min)



What Is Cyberbullying?

Define:

- **Online:** Connected to the internet
- **Cyberbullying:** Doing something on the internet, usually again and again, to make another person feel angry, sad, or scared

Discuss:

Some kids do not go online very much at all, either because of their family's rules or because they do not like it very much. Other kids do go online to do different things.

Ask:

- What do you do online, or what do you think you might like to do?
 - Students may mention activities like sending messages to friends and playing games.

Share:

Most of the time when students go online it is to do fun or interesting things. But sometimes people can be mean to each other online and this is called cyberbullying.

Ask:

- Did you ever see someone make someone else feel bad online?
 - Answers will vary. Remind students to tell what happened, but not to use real names.

Explain:

Tell students that they will be learning more about how cyberbullying occurs, and what to do when it happens to them or to someone they know.

What to Do About Cyberbullying

Discuss:

Read aloud these two scenarios and discuss them briefly with the class.

- Kyle keeps getting instant messages from someone saying mean things about him. The person who is sending the messages doesn't use a real name, but Kyle can tell the messages are coming from someone who also makes fun of him at school in gym class.
- Sasha is a new girl at school, and she's making a lot of friends. Then Sasha finds out that another girl sent around an email that had a picture of a cow with Sasha's name on it.

Next, pass out the **resource csf-screen-out-mean not found** worksheet from page 6. Read aloud the story at the top and ask students to work in pairs or groups to finish the worksheet.

Ask the class to discuss Jada's story. Tell the class there are specific steps to handling a cyberbully.

- Jada should STOP using the computer.
- Jada should TELL an adult she trusts what happened.
- Jada should not go back online or return to the pony website until an adult says it is OK.
- If Jada and Michael are good friends, Jada may want to tell Michael how his actions made her feel after she gets help from an adult.
- If Michael continues cyberbullying her, she should play with other kids who don't cyberbully others.

In general, there are four steps students should take if they or someone they know are experiencing cyberbullying.

1. Stop using the computer until it is safe.
2. Tell an adult you trust.
3. Go online only when a trusted adult says it is okay.
4. Play online only with kids who you know and are nice.

Wrap Up (15 min)

Flash Chat: What did we learn?

Ask:

- What is cyberbullying. How does it make people feel?
 - Students should recognize that cyberbullying is any kind of online behavior that makes people feel sad, scared, angry or upset.
- What four things can you do to help stop cyberbullying?
 - S. **Stop** using the computer until it is safe.
 - T. **Tell** an adult you trust.
 - O. Go **Online** only when a trusted adult says it is okay.
 - P. **Play** online only with kids who are nice.
- What is the most important thing to do if someone starts cyberbullying you?
 - Telling a trusted adult is the most important response whenever someone makes them feel sad, scared, or angry online.

Discussion

Questions to stimulate discussion include:

- What do you think happened to Jada's game?
- How do you think Jada, Kyle, or Sasha felt when these things happened to them?
- How do you know if someone is cyberbullying you?
- Why do you think it is important to stop using the computer when someone starts cyberbullying you?
 - It's possible that if students stay online, the cyberbullying may continue or get worse.

Teacher Tip:

These scenarios can be read all at once and discussed as a whole, or be read and discussed individually.

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?
- How did you feel during today's lesson?
- Write down the names of some trusted adults you can go to if you ever feel bullied.
- What are the four steps you should take if you or someone you know is being cyberbullied.

Assessment (5 - 10 min)

resource csf-screen-out-mean not found

Pass out an assessment to each student. Allow students a few minutes to complete it then review the answers (page 9 of the link above) with the class. If there's time, allow for a discussion about the questions.



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 15: Binary Bracelets

Overview

Binary is extremely important in the world of computers. The majority of computers today store all sorts of information in binary form. This lesson helps demonstrate how it is possible to take something from real life and translate it into a series of ons and offs.

Purpose

In this lesson students will learn how information is represented in a way such that a computer can interpret and store it. When learning **binary**, students will have the opportunity to write codes and share them with peers as secret messages. This can then be related back to how computers read a program, translate it to binary, use the information in some way, then reply back in a way humans can understand. For example, when we type a sentence into a document then press save, a computer translates the sentence into binary, stores the information, then posts a message indicating the document has been saved.

Agenda

Warm Up (15 min)

Vocabulary
Off and On

Main Activity (20 min)

Binary Bracelets - Worksheet

Wrap Up (5 min)

Flash Chat: What did we learn?
Journaling

Assessment (15 min)

Binary Bracelets - Assessment

Extended Learning

View on Code Studio

Objectives

Students will be able to:

- Encode letters into binary.
- Decode binary back to letters.
- Relate the idea of storing letters on paper to the idea of storing information in a computer.

Preparation

- ☐ Watch the Binary Bracelets - Lesson in Action Video.
- ☐ Gather markers for the bracelets. Other decorations like beads and pipecleaners are optional.
- ☐ Print one Binary Bracelets - Worksheet per student.
- ☐ Print one Binary Bracelets - Assessment per student.
- ☐ Make sure every student has a Think Spot Journal.
- ☐ (Optional) Write a short message on the board in binary.
- ☐ Prepare to show the Bits Versus Bytes - Student Video.

Links

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

For the Teachers

- **Binary Bracelets** - Lesson in Action Video
- **Binary Bracelets** - Assessment Answer Key

[Make a Copy](#)

For the Students

- **Binary Bracelets** - Unplugged Video (download)
- **Bits Versus Bytes** - Student Video
- **Binary Bracelets** - Worksheet

[Make a Copy](#)

- **Binary Bracelets** - Assessment

[Make a Copy](#)

- **Think Spot Journal** - Reflection Journal

[Make a Copy](#) ▾

Vocabulary

- **Binary** - A way of representing information using only two options.

Teaching Guide

Warm Up (15 min)

Vocabulary

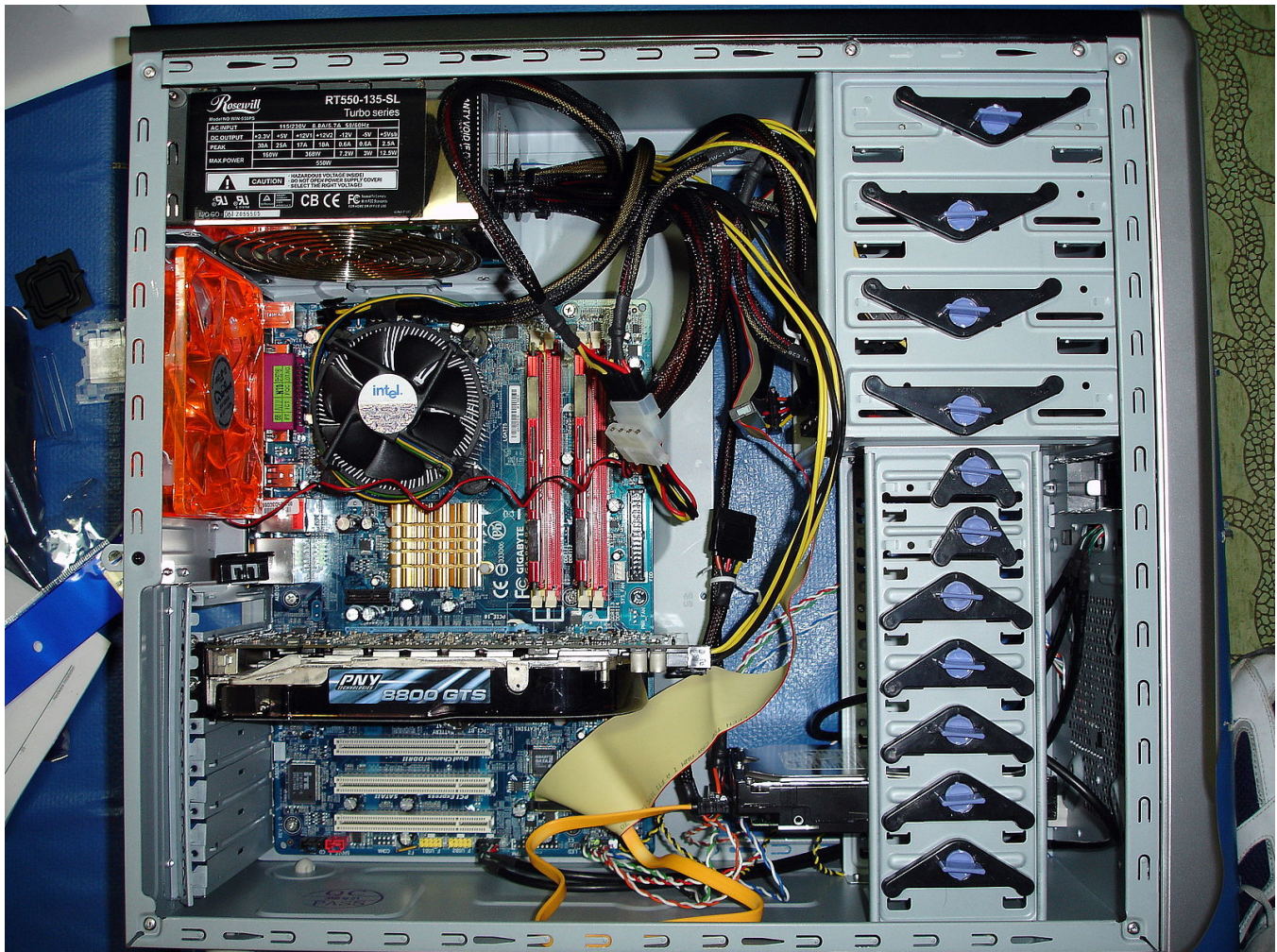
This lesson has one new and important word:

Binary - Say it with me: Bye-nair-ee

A way of representing information using only two options

Off and On

- If you've written a short message on the board in binary, call the students' attention to it and ask if anyone knows what it is or what it means.
 - Put the message aside and move on to prepping for the activity.
- You can start by asking the class if they have ever seen inside a computer.
 - What's in there?
 - This is a good place to actually show them the inside of a computer (or pictures of the inside of a computer).



- Wires carry information through the machine in the form of electricity.
 - The two options that a computer uses with respect to this electrical information are "off" and "on." Just like the lights in this room!
 - When computers represent information using only two options, it's called "Binary."
 - That theme of two options doesn't stop when the information gets to its destination.

- Computers also **store** information using binary.
 - Binary isn't always off and on.
 - Hard Disk Drives store information using magnetic positive and magnetic negative.
 - DVDs store information as either reflective or non-reflective.
 - How do you suppose we can convert real-life things that we want to store in a computer into binary?
 - Let's start with letters.
 - Use the **Binary Bracelets - Worksheet** to show how a computer might represent capital letters.
 - This is a good time to mention that each spot where you have a binary option is called a "binary digit" or "bit" for short.
 - Ask if anyone knows what a grouping of eight bits is called (it's a byte.)
 - Fun fact: A grouping of four bits is called a nibble.
 - Watch the **Bits Versus Bytes - Student Video** (~1 minute)
 - Go over a few examples of converting letters into binary, then back.
 - Afterward, write an encoded letter and give the class a few seconds to figure out what it is.
 - When the class can figure out that encoded letter on their own, you can move on to the activity.

Main Activity (20 min)

Binary Bracelets - Worksheet

You do not need to cover the whole of binary, like counting and converting numbers back and forth from decimal. This lesson is intended to be a fun introduction to how computers store information, not a frustrating lesson in bases.

💡 Lesson Tip

You know your classroom best. As the teacher, decide if students should do this individually or if students should work in pairs or small groups.

Directions:

- Find the first letter of your first name on the activity sheet.
- Fill in the squares of a bracelet to match the pattern of the squares next to the letter that you selected.
- Cut the bracelet out.
- Tape the bracelet around your wrist to wear it!
- Share your bracelet with your classmates to see if they can figure out your letter.

A	■□■□ ■■■□	N	■□■□ □□□■
B	■□■□ ■■■■	O	■□■□ □□□□
C	■□■□ ■■■□	P	■□■□ ■■■■
D	■□■□ ■■■■	Q	■□■□ ■■■□
E	■□■□ ■■■□	R	■□■□ ■■■■
F	■□■□ ■■■■	S	■□■□ ■■■□
G	■□■□ ■■■□	T	■□■□ ■■■■
H	■□■□ □■■■	U	■□■□ ■■■□
I	■□■□ □■■□	V	■□■□ ■■■■
J	■□■□ □■■■	W	■□■□ ■■■□
K	■□■□ □■■□	X	■□■□ □■■■
L	■□■□ □■■■	Y	■□■□ □■■□
M	■□■□ □■■□	Z	■□■□ □■■■

After the activity, revisit the message that was on the board and see if your class can decypher it using what they've learned.

Wrap Up (5 min)

💡 Lesson Tip

Flash Chat: What did we learn?

- What else do you think is represented as binary inside of a computer?
- How else might you represent binary instead of boxes that are filled or not filled?
- What was your favorite part about that activity?

If your class has extra budget for materials, try doing this exercise using thread (or pipe cleaners) and beads to create the binary bracelets instead of pen and paper. You can provide any combination of two colors in beads to the students, but black and white tend to be easiest, given the way that the key is done.

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?
- How did you feel during today's lesson?
- Use the activity worksheet to write out the rest of your name or your favorite word in binary.
- Imagine a world where we spoke in binary, saying "on" or "off", but nothing else. Draw two characters trying to talk to each other in binary.

Assessment (15 min)

Binary Bracelets - 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.

Binary Images

- There are several great resources on the web for taking this activity to the next level.
- If your students are interested in how images (or even music) can be represented as binary, you can find more details in Thinkersmith's **Binary Baubles**.

Standards Alignment

CSTA K-12 Computer Science Standards (2017)

- ▶ **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**.