

Lesson 6: Loops Investigate

Overview

Students practice using the for loop in order to repeatedly run pieces of code. The lesson begins with a quick investigation of an app that flips coins. After that code investigation students complete another investigation with an app that uses loops to update screen elements.

Purpose

This lesson is students primary opportunity to get hands on with loops in code prior to the Practice activity in the following lesson. Give students as much class time as you can to work through these. For this lesson it's recommended that you place students in pairs as a support and to encourage discussion about the challenges or concepts they're seeing.

Standards

Full Course Alignment

CSP Conceptual Framework

- **AAP-3** - Programmers break down problems into smaller and more manageable pieces. By creating procedures and leveraging parameters, programmers generalize processes that can be reused. Procedures allow programmers to draw upon existing code that has already been tested, allowing programmers to write programs more quickly and with more confidence.

CSTA K-12 Computer Science Standards (2017)

- **AP** - Algorithms & Programming
- **DA** - Data & Analysis

Agenda

Lesson Modifications

Warm Up (5 minutes)

Activity (35 minutes)

Wrap Up (5 minutes)

Assessment: Check For Understanding

Objectives

Students will be able to:

- Read programs that use for loops
- Understand the parts of a for loop
- Update the Boolean expression in a for loop to change how many times the loop runs

Preparation

Review the two apps that students will be investigating and the questions about them. Note that there are target responses to each of these questions on the levels.

Links

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

For the teachers

- **CSP Unit 5 - Lists Loops, and Traversals** - Slides

Teaching Guide

Lesson Modifications



Attention, teachers! If you are teaching virtually or in a socially-distanced classroom, please read the full lesson plan below, then click **here** to access the modifications.

Warm Up (5 minutes)

Discuss: *Imagine you are interested in finding out how much time it takes on average to walk from one end of your school to the next. You've decided to figure this out on your lunch break, and are able to complete the walk 20 times. What would your algorithm look like? Where could a loop show up?*

- **Note:** You do not need to write your algorithm in a programming language. You can write it out in English or in pseudocode.

Compare your algorithms with a partner and then discuss as a class.

Discussion Goal: The point of this activity is for students to see the need for a loop. Algorithms might look something like this:

- Repeat 20 times:
 - Stand at the entry doors to the school.
 - Start the timer.
 - Walk at a steady pace to the end of the hallway.
 - Turn off the timer.
 - Record the time in a notebook.
 - Walk back to the entry doors.
- Add all times together.
- Divide by 20 to find the average.

Remarks

A loop helps simplify code. Let's investigate how this looks in two different apps.

Activity (35 minutes)

Teaching Tip

For this lesson, there are two separate investigations of two different apps. Lead the class through discussions, and circulate around the classroom as students work.

As time allows, encourage students to do the Modify tasks for each level. This will give students extra practice.

Discuss: *The first app we are going to investigate uses a loops in a simulation. What is a simulation? Why are they useful?*

Discussion Goal: Simulations allow us to test solutions or run experiments in a way that is usually cheaper, safer, and often times faster. Simulations are models of the real world that allow the user to investigate hypotheses and draw conclusions. Oftentimes they involve inputting various sets of data or values to demonstrate the changing state of a phenomenon.

Remarks


Thanks for sharing your thoughts on simulations! Other examples of simulations are flight simulators which allows you to practice flying without damaging yourself or a real plane or a city building game where you can play with tweaking multiple values in the economy without real consequences.


These examples demonstrate the process of developing an abstract simulation where the specific details have been removed, or the functionality simplified.

While simulations are a useful way to model the real-world, we also need to be constantly on the lookout for bias in the real-world elements that are actually included in the simulation.


Today, we are going to look at a simple simulation.


Simulations can be helpful to run when it would take a long time or it would be too impractical to do something physically. For example, what if we wanted to see the results of a coin flip that we flipped a million times? That would take a long time to perform ourselves, but a computer can run that simulation much quicker. Let's look at an app that does just that.

 **Group:** Place students in pairs and have them navigate to the lesson on Code Studio.

 **Level 1 - Coin Flipper App:** This code investigation includes a number of steps to help students get familiar with a new app. Students run the app and discuss the following:

- What information does the user input?
- How does the app process that information?
 - What is being repeated?
 - Where is there an opportunity to use a loop?
- What information does the app output?

 **Level 2 - Coin Flipper App:** Students navigate to Level 3, run the program, watch the code run, and carefully read each individual part of the program before discussing the following:

 Run the program for 10 coin flips, 100 coin flips, and 10,000 coin flips. When do you notice it taking longer?

- On what lines of code is the program using a loop?
- Which lines of code decide how many times to flip the coin?
- What does the `++` command seem to do?



1-2

Coin Flipper App

1

2


 **Discuss:** *How does a loop help when running similar simulations?*

Discussion Goal: A loop lets us repeat code quickly to determine an answer to a question.

Inside the loop we can use random number generation to get a variety of possible answers, similar to what would happen in the real world.


Remarks

Now let's investigate an app that uses loops in a different way. This time loops are used to update elements on the screen.

 **Level 3 - Font Tester App:** Students navigate to Level 3 and run the app before discussing the following:

 **3**

Font Tester App

 **Discuss:** Find the four different for loops in the program. What do they each do? Also take a look at the names of the screen elements in Design Mode. Then navigate back to the code. How is `"text" + i` used? How does it evaluate with each round of the loop?

Discussion Goal: This is a way of doing something to a set of elements. In this case, we have five text boxes named `"text0"`, `"text1"`, `"text2"`, `"text3"`, and `"text4"`

- `"text" + i` is evaluated each round of the loop.
 - First round: `"text0"`
 - Second round: `"text1"`
 - Third round: `"text2"`
 - Fourth round: `"text3"`
 - Fifth round: `"text4"`
- So when the first round of the loop is executed, the element with the `"text0"` ID is changed to whatever information is in the for loop.

 **Do This:**


- What happens if you change the text box variable names?
- Try changing them from `"text0"`, `"text1"`, etc. to `"textbox0"`, `"textbox1"`, etc.
- Navigate back to the code. What do you need to change in the code for the program to still work?


 **Do This:**

- What happens if you change the Boolean expression `i < 5` in the for loops?
- Change all the Boolean expressions to one of the options below, run the program, and discuss. Then move to the next option and repeat:
 - `i < 4`
 - `i < 3`
 - `i <= 4`

Wrap Up (5 minutes)

 *Remarks*

 Let's review the parts of a for loop.

 **Discuss:** What aspects of working with for loops do you feel like clicked today? What do you still feel like you have trouble with?

Have students share with one another before sharing with the whole class.

Discussion Goal: Use this opportunity to address any lingering questions or misconceptions in the room. You can also use this as a source of discussion topics to kick off the following lesson. As you lead the discussion, call out the many resources students have access to help when they're getting stuck.

Remarks

Working with for loops can be a little tricky. We'll have more opportunities to practice combining loops and lists together to make even more powerful apps.

Assessment: Check For Understanding

Check For Understanding Question(s) and solutions can be found in each lesson on Code Studio. These questions can be used for an exit ticket.

Question: Think back on the Font Tester App. Can you think of an example of another app or feature of an app which would use a loop to control different elements on a screen?

 4



Check For Understanding