

# Lesson 11: Traversals Practice

## Overview

Students practice traversing lists, filtering and reducing lists, and using the data import tools. Along the way students develop debugging practices with traversals.

## Purpose

This lesson is students primary opportunity to get hands on with lists in code prior to the Make 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-2** - The way statements are sequenced and combined in a program determines the computed result. Programs incorporate iteration and selection constructs to represent repetition and make decisions to handle varied input values.

### CSTA K-12 Computer Science Standards (2017)

- **AP** - Algorithms & Programming

## Agenda

### Lesson Modifications

#### Warm Up (2 minutes)

##### Quick Warm Up

#### Activity (38 minutes)

##### Practice Time

##### Traversal Practice

##### Reduce and Filter Practice

##### App Practice

#### Wrap Up (5 minutes)

##### Synthesizing Discussion

##### Assessment: Check For Understanding - AP Practice

## Objectives

Students will be able to:

- Debug programs that use list traversals
- Write programs that use list traversals, including the filter and reduce patterns, with the support of sample code

## Preparation

- Review the levels that students will be completing with an eye for how you will encourage them to use the debugging practices emphasized in today's lesson.

## 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 (2 minutes)

### Quick Warm Up

#### *Remarks*

Today we're going to have a chance to practice programming with a lot of the concepts and patterns we've explored over the last two lessons. I encourage you to go through these with a partner, but pay close attention to what each other is doing. In our next lesson you're going to have to use a lot of these on an independent project, and these activities are good practice for what you'll find there! Alright, let's get to it!

#### Teaching Tip


**Move Quickly to the Activity:** There's a lot in the main activity of today's lesson. You may optionally wish to do a quick vocabulary review or address any questions that came up in the last lesson. Otherwise, give students more time to get hands on with some code.

## Activity (38 minutes)


### Practice Time

**Group:** It is recommended that students work in pairs for this lesson and a number of the activities feature discussion prompts. Consider using pair programming, having drivers and navigators switch every 3 minutes, not every level.

#### *Remarks*

 Today you're mostly going to practice what we've learned about programming with traversals. I'm here to help you when you need. However, I first want to remind you of the following:

- Use your debugging skills. Try to zoom in on precisely where you're getting stuck.
- Talk to your partner! That's what they're there for!
- Hover over blocks to read the documentation about how they work.
- Read the resources in the Help & Tips tab
- Talk to the group next to you. If another group asks for help make sure to take some time to talk it through with them.

 We can debug traversals by using many skills that helped us with lists like using the watch panel and console.log. Another important new debugging skill will be to actually go look at your data in the Data tab. Use console.log to make sure you're successfully getting the data you want.

## Traversal Practice

These levels have students perform simple traversals over lists that are created inside their code (not with the data import tools). In each program sample code is provided which students can use to help writing the code of their own.

- **Level 1:** traverse over a list and console.log every element
- **Level 2:** traverse over a list and console.log every element and its position
- **Level 3:** traverse over two parallel lists and print elements from both



1-3

### Traversal Practice

1

2

3

#### Teaching Tip

**Providing Support:** Circulate around the room through the lesson encouraging students to use the strategies introduced at the beginning of the lesson. Students have a number of supports at their fingertips, so a big part of your role is helping build their independence in using those resources.

## Reduce and Filter Practice

Students practice the reduce and filter patterns introduced in the previous lesson. In each program sample code is provided which students can use to help writing the code of their own.

- **Level 4:** filter a list of students to keep only those with more than 6 letters in their names.
- **Level 5:** reduce a list to find the maximum price inside of it.
- **Level 6:** filter a list of countries to find those in the "Central America" region. This program makes use of the data import tools.



4-6

### Reduce and Filter Practice

4

5

6

## App Practice

In these levels, students use the reduce and filter patterns to finish building an app.

- **Level 7:** Write code for the "reduce" pattern to calculate a student's average grade.
- **Level 8:** Write code to calculate and display the percent of each type of Air Quality Index day.



7-8


### App Practice

7

8

# Wrap Up (5 minutes)

## Synthesizing Discussion

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

**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 traversals can be tricky. We will get more practice tomorrow by making an app that uses traversals and the data tools.

---

## Assessment: Check For Understanding - AP Practice

*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:** What is stored in `studentScores` after running the program code?

```
studentScores ← [77, 32, 45, 92, 86]

FOR EACH item IN studentScores
{
  IF (item > 60)
  {
    item ← item + 5
  }
  ELSE
  {
    item ← 0
  }
}
```