

# AP with WE Service: Module Adaption for Computer Science Principles

This document can be used alongside the AP with WE Service Computer Science A Healthcare and Education Modules to adapt the programming lessons for AP Computer Science Principles (AP CSP). It intends to align these activities to the AP CSP curriculum framework and explain how they might address student focus areas. The AP CSA Healthcare and Education modules suggest three programming activities. The first uses App Inventor and fits into the AP CSP curriculum without modification. Adaptations for the other two activities, which can use any programming language, are set forth below.

## Planning for AP with WE Service for Computer Science Principles

*Follow pages 6–13 of the CSA Access to Education or CSA Access to Healthcare Module.*

### Goals

*This replaces page 14 in the CSA Access to Education or CSA Access to Healthcare Module.*

AP CSP skills and content addressed in this module:

Enduring Understandings	Learning Objectives/Essential Knowledge
EU 5.3 Programming is facilitated by appropriate abstractions	LO 5.3.1 Use abstraction to manage complexity in programs.[P3] EK 5.3.1A Procedures are reusable programming abstractions. EK 5.3.1B A procedure is a named grouping of programming instructions. EK 5.3.1C Procedures reduce the complexity of writing and maintaining programs. EK 5.3.1K Lists and list operations, such as add, remove, and search, are common in many programs. EK 5.3.1L Using lists and procedures as abstractions in programming can result in programs that are easier to develop and maintain.

## Connections to AP Computer Science Principles Focus Area

Chief Reader reports from past AP CSP administrations have identified some AP CSP Course and Exam Description content as challenging for students. Such content is referred to as a focus area. The adaptation activities described in this document address two major focus areas. When working with abstraction, students often **use procedures to group related program statements but neglect to leverage parameters to make their procedural abstractions reusable and/or hard code data or use single variables, rather than using lists as a way of creating data abstractions to store related items.**

## Part 1: Learn and Investigate

### Lesson 1: Computer Science and Positive Change

**Activity: Focusing on the topic**

**Refer to page 20 of your module. This lesson is intended to replace Lesson 1 of the CSA Access to Education or CSA Access to Healthcare Module on page 20.**

This lesson activity includes the Exploring the MLK App and can be used in AP CSP without modification.

## Lesson 2: Local and Global Access to Education / Healthcare

**Proceed to carry out Lesson 2 of the CSA Access to Education Module on page 25 or CSA Access to Healthcare Module on page 22.**

## Lesson 3: Concerns and Solutions

### Activity: Future Concerns and Solutions

**This lesson is intended to replace Lesson 3 of the CSA Access to Education Module or the CSA Access to Healthcare Module on page 27.**

Students complete methods and constructors for a Question class written in Java. It's a guide for the adaptation below, except the Java version contains a Question class, while the adaptation uses data abstraction to store questions and answers. Students create a trivia game in any programming language. **They incorporate two lists:** One stores questions, one corresponding answers. The game includes at least three procedures:

- **play** – iterates through the list of questions, calls the **getAnswer** procedure, and then displays whether the answer is correct or incorrect based on a call to the **checkAnswer** procedure.
- **getAnswer** – takes as a parameter the current question from the list of questions, displays the question to the user, and returns the user's answer.
- **checkAnswer** – takes two parameters: the answer to the current question from the list of answers and the user's answer. It then compares the two answers and returns `true` if they are the same, and `false` otherwise.

## Lesson 3: Concerns and Solutions

### Activity: Reflecting on Investigate and Learn

Students complete a class design free-response question to prepare for the AP CSA exam. In this adaptation, to prepare for the Create performance task, students write a program that demonstrates use of procedural abstraction and respond to written response prompts. They collaborate on the program writing but not on the responses.

### General Requirements

Students will complete and submit completed program code and individual prompt responses.

#### 1. Program Code

The program must demonstrate development of at least one procedure that accomplishes its intended purpose and uses one or more parameters.

#### 2. Written Responses (no collaboration allowed)

##### 2a. Provide a written response that identifies programming language and program purpose.

##### 2b. Capture and paste a program code segment of a procedure from your program.

Your procedure must contain and use one or more parameters that have an effect on procedure's functionality. Write a response saying what the procedure does and how it contributes to program functionality. Explain how the procedure helps manage the program complexity.

**Following Lesson 3, proceed to Part 2: Action Plan of the CSA Access to Education Module or CSA Access to Healthcare Module on page 41. All other elements will remain consistent following this.**