FIRIA LABS

LESSON: Flowcharts to Code

Time: 50 minutes

Overview:

This lesson is a continuation of the first lesson on the design process and flowcharts. It begins with a review of the design process and flowchart symbols. In the first lesson, students created a flowchart from Python code. In this lesson, students are given flowcharts and they create working code from them.

Objectives:

- I can explain the design process
- I can define algorithm and flowchart
- I can use the correct flowchart symbols for a specific task
- I can create a working code from a flowchart

Standards:

2-AP-10 Use flowcharts and/or pseudocode to address complex problems as algorithms.

2-AP-13 Decompose problems and subproblems into parts to facilitate the design, implementation, and review of programs.

3-AP-13 Create prototypes that use algorithms to solve computational problems by leveraging prior student knowledge and personal interests.

CSP Framework:

Computational Thinking Practices:

1.B Determine and design an appropriate method or approach to achieve the purpose.

2.A Represent algorithmic processes without using a programming language.

Key Concepts:

- Review the **design process**.
- Review the **flowchart symbols** and instructions.
- In the design process, you don't have any code yet. **Algorithms** are developed by creating a flowchart.
- **Python code** can be written based on a **flowchart**.

Preparation:

Make a copy of the assignment or put it in the LMS.
Prepare flowchart samples for students to make Python code from. This can be for individuals, pairs working together, or groups of three at the white boards. Print 2 slides per page.
Prepare any formative assessments you want to use in the wrap-up

Links:

- Assignment
- Instructions slide deck
- Flowchart symbols Kahoot
- Answers folder

Optional (instead of Kahoot!):
You can use a review worksheet from "Design Process and Flowcharts"

- Review Worksheet (for printing and completing)
- Review Worksheet (for completing online)

Agenda:

- Warm-up / Design process (10-15 minutes)
- Flowchart examples (30-35 minutes)
- Wrap-up & Assessment (5 minutes)

Vocabulary:

- Review **Design process, Algorithm,** and **Flowchart** from "Design Process and Flowcharts" lesson

Assessment:

- Daily reflection journal or Google form -- use your own link
- Assignment completion
- Demonstrate ability to create Python code from a flowchart

Teaching Guide

This lesson, like the other flowchart lesson, can go long depending on how quickly or slowly the students work. I, again, recommend that you have the students do what they can in one class period. If you have parts of lessons that haven't been finished, or haven't done some of the reviews or Kahoots or programming journal, you can schedule a "catch up" day as needed.

Warm-up / Design Process (10-15 minutes)

This extended warm-up is to review the design process and flowchart symbols from the first lesson. The assignment document isn't needed until the end of the lesson, as a review.

💡 **Teaching tip – first part of warm-up**

- You can start the lesson by asking students what they remember about the design process and flowchart symbols.
- Alternatively, you can start by showing slides #1-6. -- or go straight to the Kahoot (or other review)
- Include any student discussion about the design process or flowcharts that you feel needs to happen.

💡 **Teaching tip – second part of warm-up**

- Have students participate in the flowchart symbol Kahoot! – slide 7. This will help them review flowchart symbols associated with specific Python code. Start the Kahoot and give the join code to the students.
- Alternatively, you can use one of the review worksheets from the previous lesson, or any other review you want to with the students before starting the flowchart-to-code portion of the lesson.

Flowcharts to code (30-35 minutes)

💻 You can do this as a whole group, or with students working in pairs or groups of three. I recommend working in pairs at a computer so students can actually type the code and check that it works. Students can work in Explore mode while typing their code. They just need to click on "Select Target" and choose "USB CodeX" to run the code.

💡 **Teaching tip:**

Go over slide 8 with the students, so they have a reference for creating code from flowcharts.

Put students in pairs, and have them start working on the samples. There is no specific order, but there are three samples that are mild (one pepper), three samples that are medium (two peppers) and three that are spicy (three peppers). Students will not have time to do all 9 samples, and you shouldn't expect them to. You can suggest a mild, medium and spicy, or let them choose their levels. Students can reference the flowcharts from the slide deck, or you can print them (2 slides per page) and have them pick from the printouts.

If you want to keep track of their completed programs, you can have them copy and paste into a document. Or you can go around to the pairs and check their work. Or you can have the students check the work of each other, do a gallery walk, etc. There are many ways to manage this part of the lesson.

If time permits, students can work on the assignment document, which is a review of the Kahoot, with modifications that some of the questions are flowchart shapes, and some are Python code.

About this assignment:

- Half the questions show Python code and the student draws the correct shape and includes the instructions
- The second half shows the flowchart shape and instructions, and the student writes the Python code.

- This assignment can be completed digitally. To draw shapes, students go to INSERT–DRAWING. They can draw the shape using the shapes tool and type instructions inside, or copy and paste from below and change the instructions.
- Alternatively, you can make more space in the first half table and print the worksheet for students to complete by hand.

✅ It is up to you want you might want students to turn in for a grade. You can do most of this in class, visually. Or students can create a document to turn in with code. The assignment document is a review and can be turned in.

Wrap-Up (5 minutes)

The wrap-up can be very short for this lesson. Show slide 18, which has definitions for algorithm and flowchart. Have a discussion on anything you think students need to review for the day. Or use the time for students to complete the assignment document for review.

Formative Assessment:

- Daily reflection journal or Google form -- use your own link
- Class discussion on what they learned about the design process or flowcharts
- Assignment completion
- Update programming journal as needed
- Exit ticket

SUCCESS CRITERIA:

❏ Know the five steps in the design process in the correct order

❏ Know the symbols (or shapes) used in flowcharts, and what each shape is used for

❏ Create working code from a flowchart