Intro to Programming Part 2

Objectives: Students will devise algorithmic solutions to a variety of common real-world situations. The goal of this lesson is to help students realize that algorithms and programming are applicable to their daily lives and not solely related to computers. (Ideally, in a few weeks when we build spike counters, they will recognize that counting spikes is a problem that can be solved by a coding solution, as well as understand the logic behind how to build a spike counter.)

Concepts: Algorithmic (step-by-step) thinking, especially focusing on

- for loops
- if-then statements
- while loops
- arrays

Setup:

Materials: Whiteboards, dry erase markers, laptops with Octave installed, printouts of the individual scenarios, printouts for each student with the scenarios and solutions, signs for each of the stations

Teacher Preparation: Mentors need to be familiar with their assigned scenario and the potential problems that the students might have trouble with for that scenario. They should also be clear about the solution to the problem. In addition, it would help for mentors to be familiar with the other scenarios, so that they can try to connect ideas from the students' previous stations to the current one.

Classroom Preparation: Separate tables need to be set up for each of the scenarios. To create a scoreboard, a list of the scenarios should be written on the main whiteboard with a column for each team

In the Classroom:

(during pizza time, we will bring out the question box and ask each table to come up with one burning question they have about computers/the role of computers in their daily lives)

Warm-up Activity (5 minutes): As a class, we will quickly act out the Starburst sorting algorithm from last week, since there was no large-group review of them concepts from last week.

Lesson Introduction/Description (10 minutes): We will answer each group's question in depth, framing answers in terms of real-world situations they might encounter. From there, we will briefly describe the 6 stations and what they will be asked to do.

Activity (45 minutes): There will be 6 stations situated at 6 different tables around the room. Each station will have a real-world scenario (see attached document) in which students will be asked to solve a problem using algorithms. There will be at least one mentor at each station, in charge of introducing the scenario and guiding the students' problem solving. Mentors will also have a cheat sheet with recommended solution(s) to the problem.

Students will split up into teams (no more than 4 people), which will work independently to solve as many problems as they can within the allotted time. Using the large whiteboards, they will have to write out skeleton code that efficiently and correctly solves the problem. There will be a scoreboard on the main whiteboard with the list of stations and which teams have successfully solved each problem.

- **Next Level:** If a particular team is clearly on track to solve a problem in a short amount of time, the mentor in charge can introduce some other variables in the problem to consider.
- **Assessment:** It will be up to the mentor to determine if all members of a team have understand the solution to a given problem. Once that happens, the team will get a check mark on the scoreboard under their name.

Debrief (10 minutes): Each group will share their solution to the last problem they worked on. They will explain the starting information, what the goal was, and how they solved the problem.

Extra: If we have time, we will open up the laptops and show them some code

Schedule (expected time):

4:40-4:45 **Warm-up Activity**

4:45-5:00 **Lesson Introduction**

5:00-5:45 **Activity**

5:45-5:55 **Debrief**