Blooming Flowers

with micro:bits and Strawbees

Students will engage the design cycle while they work towards building and coding a flower structure that "blooms" via servo motor when the microbit is exposed to light.

AGE

- Gr 4+ (tested on Grades 4-7)
  - For older grades, see note in Additional Resources (tested up to Grade 12)

OBJECTIVES

Curricular content:

- Curricular competencies from Science and ADST across many grade levels
- Science 4 – Plant Sensing and Responding
- Science 5 – Simple Machines
- ADST 6/7 – Computational Thinking, Visual programming

Lesson objectives:

- Students will design and construct a simple machine that moves in response to a stimulus.
- They will consider sources of design inspiration in nature and practice using the design cycle.

MATERIALS

- Paper & pencils
- Straws & Connectors
- Per group
  - Flashlight
  - Computer (1 per group)
  - Microbit & micro-USB cord
  - Strawbees robotics board with yellow clip
  - Servo motor, plugged into "C"
- NO SCISSORS – do not allow the students to cut any of the materials

SET UP

- Organize supplies at the front of the room
- Have students log on to their computers, in groups of 2-3

ACTIVITY OUTLINE

Overview and Suggested Timeline:

Introduction	5 minutes
Light Level	10 minutes
Servo Motor	5 minutes
Strawbees	20 minutes
Reflection and Wrap-Up	5 minutes

With funding from | Avec un financement du Canada

Additionally, we gratefully acknowledge the support of the YP Heung Foundation, The McLean Foundation, and the Province of British Columbia.

Introduction

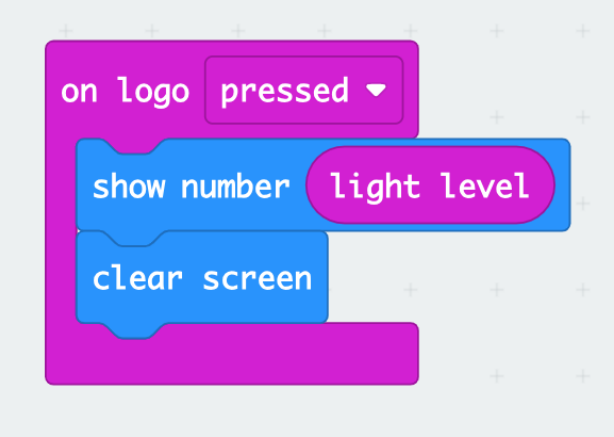
- What does "tinkering" mean?
  - Building, creating, modifying
- Where can we get inspiration when we want to create something?
  - Other people, previous inventions, nature!
  - What in nature can we be inspired by?
    - Lots of things, but also plants
- Today we will be creating machines, inspired by plants!
  - Specifically, how they move.
  - Plant growth in response to things is called tropism.
  - What do plants respond to?
    - Light, water, temperature, nutrients, touch etc.
  - We will be making machines inspired by how plants move in response to light (phototropism)!
- Design Cycle
  - As we work today, we're going to use the design cycle.
  - It starts with a **question**, then you **imagine** some possible answers, you **create** one, and then you **test** it out!
  - After **testing**, are you done?
    - No! You go back to a new **question**.
    - What is a new question you might have if your **test** is successful? What if your **test** fails?
- Brainstorming
  - Our **question** for this workshop is: what machine can we make that is inspired by how plants move in response to light?
  - We're going to start by **imagining** some possible answers. With your group, use the pencils and scrap paper to brainstorm a few ideas.
  - *Give them a minute to brainstorm*
  - Scientists are always sharing their ideas and discoveries with each other. Let's share the ideas we came up with!

Light Level

- Our machine, like a plant, is going to move in response to light! To do that, we'll have to code it. What does "coding" mean?
  - Giving instructions to machines
- We're going to code this microcontroller, called a microbit.
  - It has light sensors on the front, LEDs that can light up to output information, and buttons on either side and the touch logo on the top.
- The coding language we're using with the microbits uses pieces of code in different shapes that fit together like puzzle pieces.
  - *Show the makecode interface*
    - Virtual microbit on the left
    - Categories of code in the middle
    - Workspace on the right
- I'm going to pull 4 blocks out.
  - From "Basic"
    - Show number

Additionally, we gratefully acknowledge the support of the YP Heung Foundation, The McLean Foundation, and the Province of British Columbia.

        - Clear screen
    - From "Input"
        - On logo press (event block)
        - Light level
- Your first challenge is to figure out with your group how these fit together, and what this code is telling the microbit to do.
    - Start by **imagining** how the blocks might fit together, then build the code
    - *Have the students go to makecode.microbit.org, select "new project", then find those 4 blocks of code*
    - *Give them a couple minutes to try to fit them together like puzzle pieces*
- How do these blocks fit together?
    - *get them to tell you how to put the blocks together*
    - What do the shapes of the blocks tell you?
        - Puzzle piece shapes fit together, ovals go within ovals, the event block fits blocks inside of it but not outside
        - Event blocks tell the machine when to do something.
        - Notice that any blocks that are outside an event block are greyed out, because the microbit will never do that code.



- Let's test out our code!
    - *Show them how to plug in the microbit, download the code*
    - Send someone in your group up to get a microbit, cord, and flashlight
        - Do not shine flashlights into anyone's eyes!
    - *Give them a couple minutes to test*
    - What was the lowest number each group saw for light? The highest?
        - *Write the range down on the whiteboard, in 2 columns (get a lowest and highest from each group)*
        - *Actual light level range is 0-255*

Servo Motors

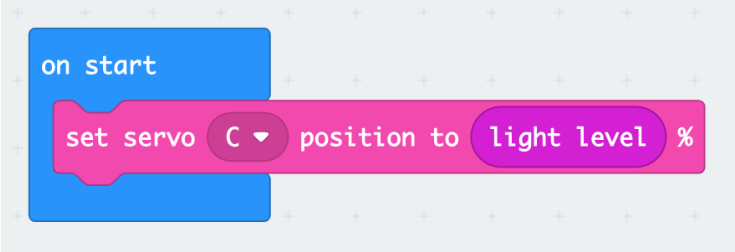
- Now let's see the motor that we will be working with! Just like the light level ranges, our servo motor has a range of movement it can do: 0%-100%.
    - *write that range on the whiteboard*
    - We can control the servo by plugging the microbit into the Robotics board connected to the servo at slot "C".
- Let's **build** code for our microbit to test out that full range of motion.
    - First, we have to get the code extension. Click on "extensions" in the categories on makecode, then search for "robotic inventions", and click on the picture with the straws.
    - We set the position of the motor within that range using the "Set servo A position to 50%", from the new "Strawbees" category
    - We need to change that to say "servo C" because our servos are plugged into slot "C"
    - We'll need 3 other blocks:
        - From "Strawbees": "Set servo A position to 50%"
        - From "Input": "On button A press" x2 (event block)

Additionally, we gratefully acknowledge the support of the YP Heung Foundation, The McLean Foundation, and the Province of British Columbia.

- Your **question** is to figure out how to build this code and fit these blocks together to be able to toggle the motor back and forth from 0-100. Remember to start by **imagining** with your group, then **create** the code.
  - *Give them a couple minutes to put the code together.*
  - How do these blocks fit together?
- Let's **test** it out! Download the new code onto your microbit.
  - Send up someone in your group to get a robotics board. Plug the microbit into the robotics board.
  - IMPORTANT: always hold both the robotics board and the servo motor, never let one dangle from the other (this will damage the wires)
  - Turn on the robotics board (button on top right)
  - **Test** the code by pressing buttons A & B!

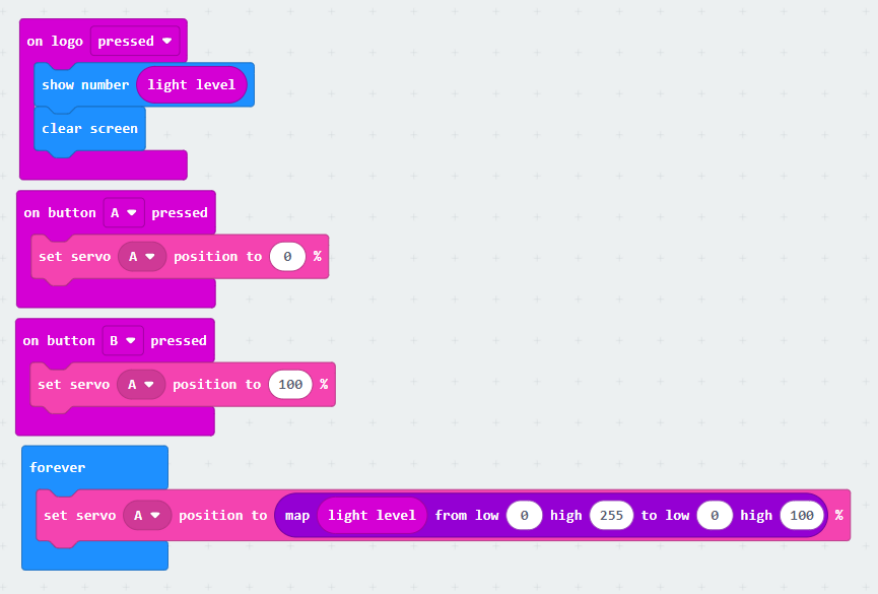
Strawbees

- It's time to use that servo movement to make a machine.
- Your **question** is how to use the movement of the servo to make the machine mimic one of the ways a plant would move in response to light. Your **creating** supplies are these straws and connectors.
  - *show a few ways to connect straws and strawbees, how to bend the strawbees, how to attach to the servo motor arms*
  - These straws are not drinking straws! They have been touched by many other students. Please don't put them in your mouth.
- You will now have a few minutes to start **creating** your machine. Remember to use the design cycle: **imagine** what you want to do, **create** it, then **test** it! Then come up with a new **question** based on the result of your **test**.
  - Start with the 0%/100% toggle with the buttons, to see what the movement will be. We'll add in the light level in a few minutes!
  - *Give them a few minutes to start building their machines.*
- We've had some time to start **creating** and **testing** the movement of our machines. Now it's time to incorporate the light sensors.
  - We want to movement to happen whenever the light changes, all the time. Here are the first few blocks we need:
    - "Basic": "Forever" (event block)
    - "Strawbees": "Set servo A position to 50%"
  - How can we set the position, not to a specific position but to change with the light level? What should we put where it says the percent?
    - "Light level" block (from Input)
  - Can anyone guess the problem with this? What position does the servo go to if the light level is:
    - 0? 0%
    - 100? 100%
    - 150? Still 100%, because it can only go up to 100%



Additionally, we gratefully acknowledge the support of the YP Heung Foundation, The McLean Foundation, and the Province of British Columbia.

- We need to change the range of the light level into the same as the servo range. We'll use a block from the "Math" category: "Map 0 from low 0 high 1023 to low 0 high 4"
  - Except, 0-1023 & 0-4 aren't the right ranges. You need to change those numbers to the range of light level and the range of the servo motor, and then put the blocks together.
  - Once you've **created** the code, download it onto the microbit and **test** it out!
  - *Give them a few more minutes to build*



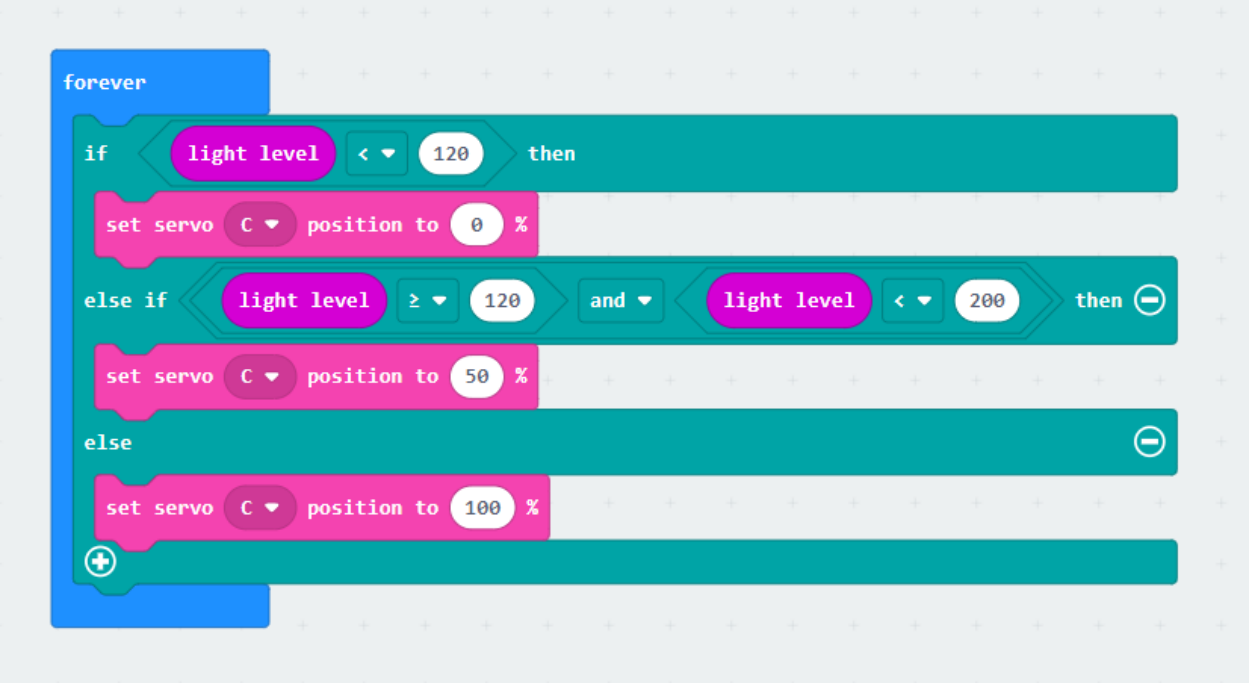
- It's time to do get some more inspiration! We can sometimes be inspired by other people and things they've made. Everyone, stand up, and for one minute we are going to walk around and take a look at what other groups have made.
  - *After 1 minute:* Now you can go back to your own workspace. You have a few more minutes to build! Maybe you want to change your machine if you've been inspired by someone else. What new **questions** do you have?
- Give them a few minutes to finalize their creations.

Optional Extensions

- Other methods to have the servo respond to the light level:
  - Math
    - Do some math with the students. They have to figure out how to convert the 0-255 range of the light level to the 0-100 range of the servo motor.
    - Once they've figured out what they have to divide the light level by, they can use the "0 ÷ 0" math block

  - Conditionals
    - You can make the servo motor move to a pre-set angle depending on a condition. This uses several blocks from the "Logic" category.

Additionally, we gratefully acknowledge the support of the YP Heung Foundation, The McLean Foundation, and the Province of British Columbia.



  - The picture below is an example of how to do that.
- The microbit can respond to several different stimuli. If students want to experiment, direct them to try having the servo move their machine in response to other things, like sound or which way the microbit is facing.
  - The relevant blocks are in the "Input" category, but to use some of them you may have to also use the conditionals from "Logic".
  - Sound level works very similarly to light level
- As a maker extension, have the students work with other groups to combine their machines.
  - Make a machine with 2-3 moving parts, either using other microbits or by plugging servos in to one microbit in the slots A & B in addition to C.
- For the moving part of the machine, challenge the students to make a joint (where the movement isn't one-to-one with the servo, but rather the servo moves one thing which moves another).
  - Can they create a piece that moves up-and-down, or side-to-side, even though the servo is only turning in a semicircle?
- Have the students research an invention that has been inspired by plants.

Reflection and Wrap-Up

- Put away supplies before asking reflection questions
- Share challenges and successes from the activity with each other
- Ask the reflection questions (below)

REFLECTION QUESTIONS

- Why is the testing part of the design cycle important?
- How does knowing coding help us to create and tinker?
- Where else could we use machines that respond to light? Other stimuli?
- Why is nature a valuable inspiration for tinkering and innovation?

Additionally, we gratefully acknowledge the support of the YP Heung Foundation, The McLean Foundation, and the Province of British Columbia.

TROUBLESHOOTING TIPS

- Familiarize yourself beforehand with loading code from makecode.microbit.org onto a microbit,
  - See microbit.org/start for instructions and a guide to using microbits.
  - You will need to show students how to access the code, and how to load the code onto the microbits
  - The students must remove the microbit from the robotics board before plugging it in to the computer.
- Also familiarize yourself with the various ways to use the straws and connectors
  - Make an account at classroom.strawbees.com for all the strawbees resources
- Use slot "C" to plug the servo in, otherwise students get confused about buttons A & B and change servo to "A" & "B"
- If the servo isn't working, check:
  - Servo must be plugged in to "C", with the black cord at the arrow
  - Code must say "servo C"
  - Robotics board must be turned on
  - If it's still not working and the code looks right, just redownload the code onto the microbit and try again

ADDITIONAL RESOURCES

- Older students may benefit from going a bit less step-by-step than this lesson. For them, still work with the different lesson sections, but try just showing them the blocks they'll be using and let them experiment with how they fit together, trying to figure out how to make the code do what they want.
  - Use the Optional Extensions for the lesson for older students, or groups who solve the challengers a bit faster and need something new to try

Additionally, we gratefully acknowledge the support of the YP Heung Foundation, The McLean Foundation, and the Province of British Columbia.