CANADA LEARNING CODE

Deconstructing Code

By: Kassandra Lenters + Jaci Banton
Duration: 20 minutes

LEVEL	SUBJECTS	PROVINCES / TERRITORIES	TOOL
Grade 4-6, 7-8	Language Arts	Across Canada	Unplugged

Overview

In this activity, students practice interpreting code by acting out printed algorithms in small groups, then work to debug (or fix) the code together as a class.

Key Coding Concepts

- Events
- Loops
- Conditional Statements

Terminology

Debugging

Finding problems in code and solving them.

Prep Work

- Print/scan the Decode Sheets for the main activity: http://bit.ly/scratch-decode (1 per group of 6 students).
- Cut out the decode sheets ahead of time.
- This activity requires some space to move around.
- The closure of this activity requires a computer with access to the internet (or the Scratch offline editor installed).

Lesson

Activity:

1. Divide the class into groups of 5-6.
2. Give each group a stack of the cut out decode sheets.
3. Instruct each learner to take one paper (#1-6) - No peeking!
4. Learners will try to 'decode' or interpret the code on their paper by acting it out.
   *Note: Group participation may be required

5. Instruct learners to begin with a high-five (to replace the “When sprite is clicked” or “When space key is pressed” events).
6. If learners try acting it out but get stuck, they can ask their group for help.

Closure:

1. All of the code that we looked at are originally from a project in Scratch!
2. Check out the original project: https://bit.ly/de-code-solution
3. Click “See Inside”
4. Go through each number: Select the character from the Sprites area to see the code, then do what it says (e.g. click on it or press the space key).
5. For each, ask the learners that were assigned this number if this is what they ‘decoded’ when they acted it out.
6. Point out **Events** (Orange ‘hat’ blocks that tell us when to begin), **Loops** (yellow “C” shaped blocks that make things happen more than once), and **Conditional statements** (yellow “IF, THEN” blocks that only happen IF something is true).
7. Note: One of the sprites doesn’t do anything when we click on it (#4) - Why not? Try to debug (fix it) together. What does this tell us about computers? (They take things very literally! We need to give them super clear instructions).

Assessment

Learning Outcomes

- Computers need very clear instructions in a specific order (or sequence).
- Events tell the computer when to start executing code.
- Loops make things happen more than once.
- Conditional statements control IF something happens or not.
- Sometimes computers don’t understand our instructions, and we need to debug (or fix) the code.

Extensions

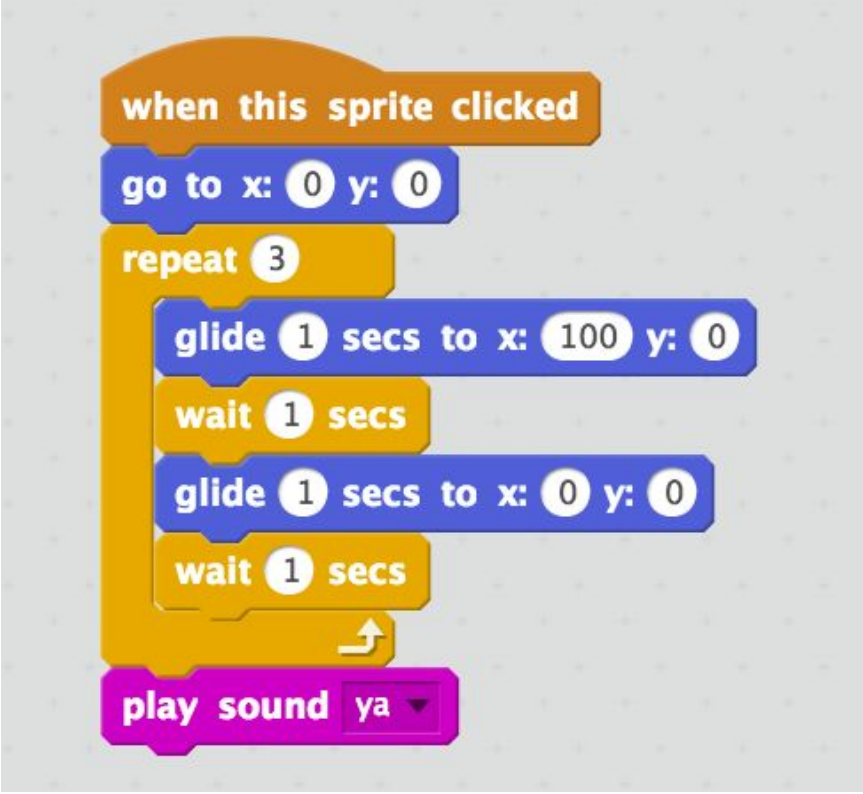
If you have access to computers, have learners create their own algorithms in Scratch, then have a partner try to decode and act out their instructions before seeing how the sprite interprets the code.

Alternatively, print out larger paper versions of 5-10 different blocks from Scratch and have learners put them together like puzzle pieces. Then, have another learner decode their algorithm by acting it out.

#1


when this sprite clicked
say Lifetime supply of pizza! for 2 secs
repeat 4
move 3 steps
turn 90 degrees
change size by -2
set size to 100 %

#2


when this sprite clicked
go to x: 0 y: 0
repeat 3
glide 1 secs to x: 100 y: 0
wait 1 secs
glide 1 secs to x: 0 y: 0
wait 1 secs
play sound ya

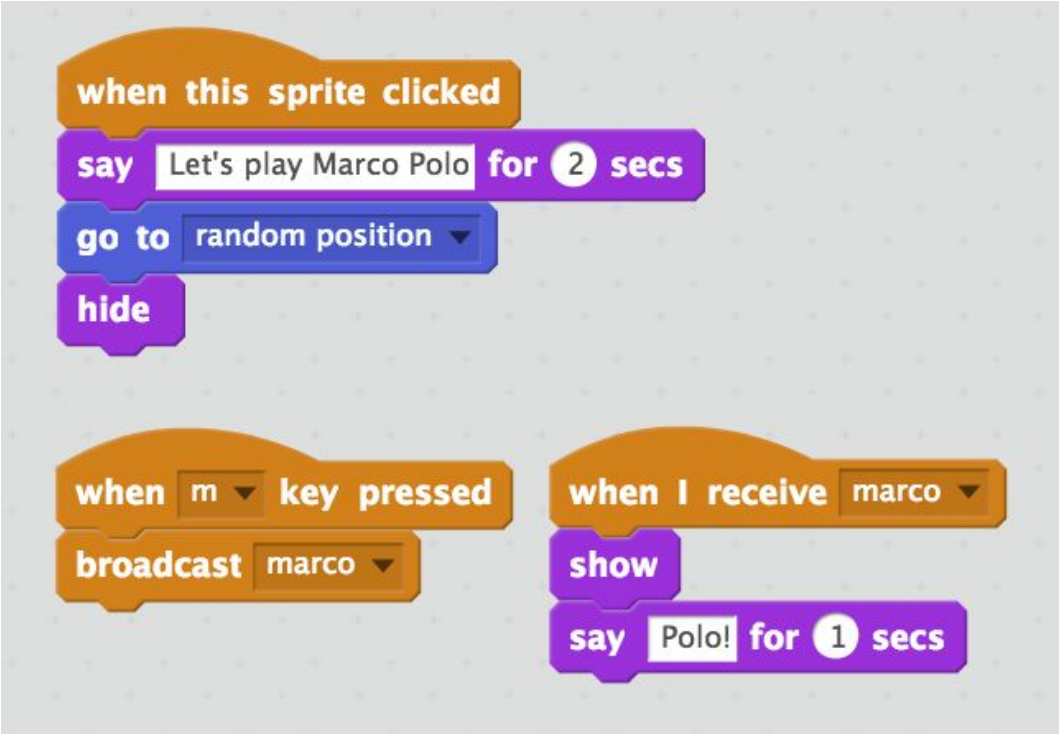
#3


when this sprite clicked
set year to 2019
if year = 2019 then
repeat 3
turn 360 degrees
say WOO! for 1 secs

#4


when this sprite clicked
set counter to 0
repeat counter
move 3 steps
turn 180 degrees
wait 1 secs
change counter by 1
if counter > 5 then
stop all

#5


when this sprite clicked
say Let's play Marco Polo for 2 secs
go to random position
hide
when m key pressed
broadcast marco
when I receive marco
show
say Polo! for 1 secs

#6


when space key pressed
ask What do you call spaghetti that doesn't belong? and wait
if answer = an impasta then
say Have you hear this joke before? for 2 secs
else
say an impasta! for 1 secs