(http://csmatters.org) 2 - 10

0b10 - 0b1010

Iteration: For Loops

Unit 2. Developing Programming Tools

Revision Date: Jul 22, 2019
Duration: 1 50-minute session

Lesson Summary

Summary

In this lesson, students will be introduced to the concepts of iteration and `for` loops.

Outcomes

- Students will work through a guided tutorial on for loops while being introduced to using turtle graphics in Python.
- Students will practice writing programs using for loops and turtle graphics.
- Students will journal as a reflective tool to make a personal connection between iteration and their personal life.
- Students will describe how computation facilitates the creation and modification of computational artifacts with enhanced detail and precision.

Overview

1. Getting Started (5 min)
2. Introduction of Content (40 min)
    1. Activity [10 min]
    2. Journal [5 min]
    3. Activity [10 min]
    4. Individual Coding [15 min]
3. Wrap Up (5 min)

Note: Turtle graphic examples in this lesson work with the community version of the PyCharm IDE and Python 3.4.1.

Learning Objectives

CSP Objectives

- *EU AAP-2 - The way statements are sequenced and combined in a program determines the computed result. Programs incorporate iteration and selection constructs to represent repetition and make decisions to handle varied input values.*
    - LO AAP-2.J - Express an algorithm that uses iteration without using a programming language.
    - LO AAP-2.K - For iteration: a. Write iteration statements. b. Determine the result or side-effect of iteration statements.

Math Common Core Practice:

- MP1: Make sense of problems and persevere in solving them.
- MP2: Reason abstractly and quantitatively.
- MP4: Model with mathematics.
- MP5: Use appropriate tools strategically.
- MP6: Attend to precision.
- MP7: Look for and make use of structure.
- MP8: Look for and express regularity in repeated reasoning.

Common Core ELA:

- WHST 12.2 - Write informative/explanatory texts, including the narration of historical events, scientific procedures/experiments, or technical processes
- WHST 12.4 - Produce clear and coherent writing in which the development, organization, and style are appropriate to task, purpose, and audience

NGSS Practices:

- 3. Planning and carrying out investigations
- 5. Using mathematics and computational thinking
- 6. Constructing explanations (for science) and designing solutions (engineering)
- 8. Obtaining, evaluation, and communicating information

Key Concepts

Students must understand that programs use the concept of iteration to perform repeated tasks.

Essential Questions

- How can computing and the use of computational tools foster creative expression?
- How can computing extend traditional forms of human expression and experience?
- How are programs used for creative expression, to satisfy personal curiosity or to create new knowledge?
- How do computer programs implement algorithms?
- How do people develop and test computer programs?
- Which mathematical and logical concepts are fundamental to computer programming?

Teacher Resources

Student computer usage for this lesson is: **required**

In the Lesson Resources folder:

- Turtle Graphics Guided Activity

Required materials:

- 5-10 objects that can be stacked (lego, duplo blocks, plastic cups…) for the first guided activity

Useful additional resources:

- How to Think Like a Computer Scientist (Ch. 4 Python Turtle Graphics (http://interactivepython.org/runestone/static/thinkcspy/PythonTurtle/intro-HelloLittleTurtles.html))
- How to Think Like a Computer Scientist (Ch. 8 Iteration Revisited (http://interactivepython.org/runestone/static/thinkcspy/MoreAboutIteration/intro-IterationRevisited.html))
- Python for Everybody (http://do1.dr-chuck.com/pythonlearn/EN_us/pythonlearn.pdf)
- CS1 Python Programming Projects Archive http://www.cse.msu.edu/~cse231/PracticeOfComputingUsingPython/ (http://www.cse.msu.edu/~cse231/PracticeOfComputingUsingPython/)

Lesson Plan

Getting Started (5 min)

Journal: Think about events in your life that require a repeated action. They could be something simple such as eating a bowl of cereal. List two events in your life that require an action to be repeated. What is the action? What prompts the need for the action to happen? How often does the action happen?

Note: Students will extend their reflections later in the lesson.

Introduction of Content (40 min)

Guided Activity: Physical representation of iteration [10 min]

Materials: 5-10 objects that can be stacked (lego, duplo blocks, plastic cups…)

The activity: Place the objects on a table.

1.Say: “At the conclusion of this activity, all of the objects will be stacked.”

Chose one object to begin with.

2. Say: “I will start with this object and I will continue to stack until there are no single objects left on the table. How many times do you think I will stack an object? Why?”

3. Ask: "Are there any single objects on the table?”

Students should answer yes. Stack one object on your beginning object

4. Ask “ Are there any single objects on the table?”

Students should answer yes. Stack one more object on your started stack. Continue to ask if there are any single objects on the table until the stack is completed and there are no more single objects on the table. Keep a tally of how many times you repeated the process.

5. Ask: “How many times did we repeat the process? Did your prediction match the result?"

6. Explain to the students how the activity represents the concept of iteration and continue into the discussion of iteration.

7. Ask: “How much better is a computer at providing accurate and detailed results of a very long, repetitive process?" "Why is a computer able to be more precise and detailed?" Computers are used in assembly lines and other places with lots of repetition because they can be almost perfectly accurate thanks to sensors that can be calibrated to the millimeter, and without needing to take a break.

Journal [5 min]

Choose one of the events you wrote about in your previous journal entry. Take a moment to write the pseudocode for the repetitive action associated with that event.

Note: Check for understanding while students are working.

Example: Eating a slice of pizza:

```
While pizza on plate
    pick up from plate
    take a bite
    place on plate
    bite is consumed
    loop
```

Guided Activity (for loops, how for loops use lists, range, turtle graphic) [10 min]

This guided activity introduces students to `for` loops using turtle graphics.

(See handout in Lesson Resources folder: Turtle Graphics Guided Activity: The for loop)

Individual Coding Activity [15 min]

Give students the following code stem. Have the students alter the code to perform the listed tasks.

Code Stem:

```
import turtle    # Allows us to use the turtles library
```

```
window = turtle.Screen()  # Creates a window to display graphics

bob = turtle.Turtle()   # creates a turtle named bob

#Write your code here

window.exitonclick()      # Exits the window when clicked
```

1. Have the turtle draw a triangle using a turtle

2. Now that you know how to add turtles and program them to draw lines repeatedly, use your imagination and creative ability to create your own picture using multiple `for` loops and turtles.

Wrap Up (5 min)

Journal: In your journal summarize the process you used to create your picture. What problems did you encounter? What concepts do you need clarified?

Options for Differentiated Instruction

Students can be given a copy of the guided activity handout to follow along.

Evidence of Learning

Formative Assessment

a variety of checking for understanding techniques

- temperature checks
- teacher review student's code
- thumbs up/ thumbs down
- questioning throughout the lesson (whole group / small group / individual)

quick quizzes

peer review

interactive journaling

Summative Assessment

Students will use for loops and turtle graphics to create graphic representations of iteration. They modify a code stem using turtle graphics to:

1. draw a triangle
2. draw a picture using multiple for loops and turtles.

(http://www.umbc.edu/) (http://www.umd.edu/)

(http://www.nsf.gov/)

Authored by: CS Matters in Maryland
Website: csmatters.org (http://csmatters.org)
Email: csmattersinmaryland@gmail.com (mailto:csmattersinmaryland@gmail.com)