(http://csmatters.org) 4 - 3

0b100 - 0b11

Using Data and Simulations

Unit 4. Data Acquisition

Revision Date: Feb 06, 2020
Duration: 1 50-minute session

Lesson Summary

Summary

Students will formulate a hypothesis, run simulations, and analyze the results to determine what needs to be modified in their hypothesis and/or the simulation itself.

Outcomes

- Students will identify and create real-world examples of models and simulations.
- Students will be introduced to using random numbers both in Python and the exam reference sheet.
- Students will use models and simulations to generate new knowledge and to formulate, refine, and test hypotheses.
- Students will use simulations to test hypotheses without the constraints of the real world.

Overview

1. Getting Started (10 min)
2. Guided Activities (35 min)
    1. Activity 1 - Rolling Dice Simulation, materials - large paper and markers or other method to share group work [15 minutes]
    2. Activity 2 - Using a Simulation to Test a Hypothesis [20 minutes]
3. Wrap Up (5 min)

Source

The coin flipping extension is based on a CS10K lesson: https://sites.google.com/site/mobilecsp/lesson-plans/lp-coinflip-miniprojects (https://sites.google.com/site/mobilecsp/lesson-plans/lp-coinflip-miniprojects)

Learning Objectives

CSP Objectives

- *EU DAT-2 - Programs can be used to process data, which allows users to discover information and create new knowledge.*
    - LO DAT-2.D - Extract information from data using a program.
    - LO DAT-2.E - Explain how programs can be used to gain insight and knowledge from data.
- *EU AAP-1 - To find specific solutions to generalizable problems, programmers represent and organize data in multiple ways.*
    - LO AAP-1.A - Represent a value with a variable.
    - LO AAP-1.B - Determine the value of a variable as a result of an assignment.
- *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.H - For selection: a. Write conditional statements. b. Determine the result of conditional statements.
- *EU AAP-3 - Programmers break down problems into smaller and more manageable pieces. By creating procedures and leveraging parameters, programmers generalize processes that can be reused. Procedures allow programmers to draw*

upon existing code that has already been tested, allowing them to write programs more quickly and with more confidence.

  - LO AAP-3.A - For procedure calls: a. Write statements to call procedures. b. Determine the result or effect of a procedure call.
  - LO AAP-3.E - For generating random values: a. Write expressions to generate possible values. b. Evaluate expressions to determine the possible results.
  - LO AAP-3.F - For simulations: a. Explain how computers can be used to represent real-world phenomena or outcomes. b. Compare simulations with real-world contexts.

Math Common Core Practice:

- MP2: Reason abstractly and quantitatively.
- MP3: Construct viable arguments and critique the reasoning of others.

Common Core Math:

- F-BF.1-2: Build a function that models a relationship between two quantities
- S-ID.1-4: Summarize, represent, and interpret data on a single count or measurement variable

Common Core ELA:

- RST 12.7 - Integrate and evaluate multiple sources of information presented in diverse formats and media
- RST 12.8 - Evaluate the hypotheses, data, analysis, and conclusions in a science or technical text
- RST 12.9 - Synthesize information from a range of sources

NGSS Practices:

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

Key Concepts

Students will be able to:

- identify and create real-world examples of models and simulations.
- use models and simulations to generate new knowledge and to formulate, refine, and test hypotheses.
- use simulations to test hypotheses without the constraints of the real world.

Essential Questions

- How can computational models and simulations help generate new understanding and knowledge?
- How can computation be employed to help people process data and information to gain insight and knowledge?
- How are algorithms implemented and executed on computers and computational devices?
- How are programs developed to help people, organizations or society solve problems?
- 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**

The PowerPoint "Using Data and Simulations" can be found in the Lesson Resources folder.

Penny Bias article to go with lesson extension: http://mathtourist.blogspot.com/2011/02/penny-bias.html (http://mathtourist.blogspot.com/2011/02/penny-bias.html)

For the Monty Hall Problem extension:

Online simulation of the problem: http://math.ucsd.edu/~crypto/cgi-bin/MontyKnows/monty2?1+17427 (http://math.ucsd.edu/~crypto/cgi-bin/MontyKnows/monty2?1+17427)

There are several videos on YouTube demonstrating and explaining the Monty Hall Problem.

An animated video: https://www.youtube.com/watch?v=mhlc7peGlGg (https://www.youtube.com/watch?v=mhlc7peGlGg) length is 5:48

Live action video: https://www.youtube.com/watch?v=4Lb-6rxZxx0 (https://www.youtube.com/watch?v=4Lb-6rxZxx0) length is 5:30

There is sample code for the die Python program in the Lesson Resources Folder called 4-3 Sample Code.py

Lesson Plan

Getting Started (10 min)

Journal:

- Models and simulations are simplified representations of more complex objects or phenomena.
  - What is a complex, everyday phenomena that is studied using models and simulations?
- Models often omit unnecessary features of the objects or phenomena that are being modeled.
  - What features might a flight simulator program leave out to make the program easier to use and run faster?
- Simulations mimic real-world events without the cost or danger of building and testing the phenomena in the real world.
  - What is something random that could be tested more easily with a computer than by building the real thing?

Have students share their answers with the class.

Guided Activities (35 min)

Activity 1 (15 min) - Designing a Simulation (rolling dice)

1. Explain that many simulations include randomized events to simulate reality. Random number generators can be used to simulate the variability that exists in the real world. Using random number generation in a program means each execution may produce a different result and that allows the program to simulate a large number of seeming random events.
   - What unnessecary details can be omitted if you need to create a random outcome?
   - [Possible answers: you only need to choose a random number and then use selection to choose which block of code to execute as a result]
2. Arrange the class into small groups (two or three).
3. The groups are to develop an algorithm and write a program to simulate rolling a standard six-sided die.
   - The program should ask for the number of rolls and display the results (number of each possible outcome that was rolled).
   - **Note: The students will need to use the randint function to produce a series of "random" differering values.**
     - `import random` at the top of the code
     - `random.randint(min,max)` returns a 'random' integer between the min and max values (inclusive).
     - The exam reference sheet includes a RANDOM(a, b) procedure which returns a random integer between a and b, inclusive.
   - Remind students to use functions where appropriate.
   - Students are encouraged to write their plan (flow chart or pseudocode) on large paper or some other method to enable later sharing with the class.
4. Have a representative from each group describe their plan for the algorithm. (After the first group, subsequent groups could highlight similarities and differences.)
5. Next, have each group discuss and implement changes to their algorithm and programs to simulate rolling two dice. (When providing directions, ask the class how using functions in the previous exercise would make this change easier to implement.)
6. As a class, discuss how many times the program should "roll" the dice. (Be sure students note that every possible value should be displayed, especially upper and lower bounds.)

Activity 2 (20 min) - Using a Simulation to Test a Hypothesis

1. Each group should develop a hypothesis to answer the question: Can a twelve-sided die be used in place of rolling two six-sided dice? Each group should agree on a hypothesis and write it down.

2. The groups already have a design for rolling two six-sided dice. They should now design an algorithm for rolling a twelve-sided die. (This process can involve creating a new function or modifying the current function's parameters to accept the number of sides.)
3. The students should be paired up and begin writing the code necessary to test the hypothesis.
4. Once they finish the code, students should run both simulations to compare results. **Note: Students need to save their code. They will be using it in the next lesson's homework.**
5. As students complete their simulations, if possible, have each pair share their data with the class (in a spreadsheet displayed by the teacher's computer, or on the white board, or large paper).
6. As data is collected, students should determine whether their hypothesis was correct or whether it needs to be modified.
7. Analyze with the class if the simulations themselves were implemented correctly. (Does the data represent the expected theoretical outcomes? How can you modify the program to correct any errors?)

Wrap Up (5 min)

Ask the class: What are the advantages/disadvantages of using a program vs. actual dice? How quickly can the computer generate thousands of test cases? Can the computer be used to analyze the test cases as well as to generate the random numbers?

Journal: Summarize how a program can be used as a simulation to test a hypothesis.

Options for Differentiated Instruction

Students can be provided with the code for a function to simulate rolling one die and use it to develop the rest of the program.

After the first group activity, the teacher can swap a student from each group to allow different input into the next group activity.

Extension #1 - Penny Flipping

This extension is based on advanced mini project # 4, which can be found here: https://docs.google.com/a/smcps.org/document/d/1AKHpiQ87bE4W1YzHlAFh2uNAHuEtdMOCQVV6HfxfDzc/edit (https://docs.google.com/a/smcps.org/document/d/1AKHpiQ87bE4W1YzHlAFh2uNAHuEtdMOCQVV6HfxfDzc/edit)

Students read an article about the 'randomness' of flipping a penny: http://mathtourist.blogspot.com/2011/02/penny-bias.html (http://mathtourist.blogspot.com/2011/02/penny-bias.html)

Next, students should hypothesize the results of lining up 10 pennies on edge and knocking them over (as described in the article). Students need to determine how many times to run the experiment, collect data, and analyze the results.

Students should work in pairs to write a computer simulation for the penny experiment. (Note: this is a program based on experimental data, not theoretical.)

Discuss as a class the validity of the simulation written. Can this simulation be used for other coins?

Extension - The Monty Hall Problem

In the game show "Let's Make a Deal", the original host was Monty Hall. Onvery show, Monty would present a player with three doors or curtains to choose from. The contestant was asked to choose a door in search of a prize. After making a selection, Monty Hall would open one of the doors not selected by the contestant to reveal a non-prize (perhaps a goat). Then Monty would ask if the contestant wanted to change their choice.

After explaining the show to the class ask, "Should the contestant change?" Students should propose a hypothesis.

Have the students design a simulation to test their hypothesis (discuss what is the data collected and the number of times the simulation should run to collect data). After running the simulation, students should evaluate their hypothesis and determine whether it needs to be modified or whether the simulation needs to be modified.

If Monty Hall had four doors, what should the contestant do?

What should the contestant do if they know that Monty does not know what is behind each door?

Online simulation of the problem: http://www.math.ucsd.edu/~crypto/Monty/monty.html (http://www.math.ucsd.edu/~crypto/Monty/monty.html)

There are several videos on YouTube demonstrating and explaining the Monty Hall Problem.

An animated video: https://www.youtube.com/watch?v=mhlc7peGlGg (https://www.youtube.com/watch?v=mhlc7peGlGg) length is 5:48

Live action video: https://www.youtube.com/watch?v=4Lb-6rxZxx0 (https://www.youtube.com/watch?v=4Lb-6rxZxx0) length is 5:30

Evidence of Learning

Formative Assessment

Review student journal entries and class discussions to determine students' understanding of simulations, a hypothesis, and the ability to determine a method to test a hypothesis.

Summative Assessment

Describe an algorithm to simulate drawing an ace of any suit from a standard deck of cards.

Make a hypothesis about drawing cards from a standard deck of cards and determine how to collect data to answer your hypothesis.

(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)