Lesson Plan: Algorithms and Automation—A Compliment Generator

Original Curriculum

CT Focus: Automation	 **https://goo.gl/ZfsbTD**
Cross-Curricular Ties: English and Language Arts	
Age Range: 10–16	
Duration: 45 minutes	
Scan the QR Code or type the URL to see a teacher-facing YouTube video on algorithms.	

Overview

With this activity, students will learn about the relationship between algorithms and automation by creating a compliment generator. Students will figure out how to break sentences into chunks (beginning, middle, end) and then how to mix and match those chunks into new sentences. Once the procedure has been identified, students will write the algorithms for their generators, so that the procedure can be automated.

Vocabulary

Algorithm: A list of steps that can be followed to carry out a task.

Automation: Having a machine (such as a computer) do work for us, so that we don't have to do it ourselves.

Pseudocode: Instructions that look like they could be a computer program, but they are easier to read and don't necessarily follow rules of any specific programming language.

Lesson Objectives

Students will be able to:

- Break sentences apart into appropriate sections for randomization
- Compose sentences from random pieces
- Write an algorithm that explains the actions that the student's "machine" should take to automate the sentence-building procedure.

Materials and Resources

- Paper
- Pencils
- Whiteboard or projector
- Paper cups (three per group)

Lesson Plan: Algorithms and Automation—A Compliment Generator

Preparation

1. Read the lesson and review the video linked with a QR code at the beginning of this lesson for better understanding of "algorithms" and "automation."

Activity

Step 1. Introduction—This lesson can do with a little prep around algorithms. Since algorithms and automation are so closely related (and both start with the letter *a*), it's easy to confuse the two.

First, have a chat about algorithms.

- Has anyone ever heard of an algorithm before?
    - What is it?
    - What do you think it might be?
- An algorithm is just a list of instructions that you can follow to finish a task. It's like a recipe or a step-by-step tutorial for modding your video game.
    - What else can you think of that might be an algorithm?

Now, take some time to come up with an algorithm for something students do every day, like getting ready for school or cleaning the kitchen.

Allow your class to come up with a couple more algorithms, until you're comfortable that they understand what an algorithm is. You might want to take the opportunity to let the students create one with a neighbor and share it with the class.

Algorithms let us prepare things for automation.

- Has anyone heard the term *automation* before?
    - What does *automation* mean to you?
    - Doing something automatically?
    - Having a machine do our work?
- Essentially it's having a machine or tool do work for us so that we don't have to do it ourselves. What do you use daily that automates things for you?
    - A calculator
    - A car
    - A printer
- What does it automate, do you suppose?

Step 2. Program and run—Once you've gone through that thought exercise, you're ready to get your students thinking about setting a task up to be automated. Let them know that they will be gathering into small groups to prepare a task for automation. One person will be the "machine," and everyone else will be part of the programming team. (If you have very young students, you will likely want to have the whole class program while you act as the machine.)

Now, time for the activity!

Share some compliments with your class that fit the same structure as these:

"You are a beautiful person."

"He is a kind human being."

"She is a strong role model."

You can deliver these authentically to your students, but be sure to write them down on the board or overhead screen so that you can dissect them later.

- What did all of those compliments have in common? (The structure)
- What if we wanted to mix these up?
  - "You are a kind role model."
  - "He is a strong person."
  - "She is a beautiful human being."
- Where can we break these sentences up to be able to mix them all around?
  - "You are"
  - "He is"
  - "She is"
  - "a kind"
  - "a strong"
  - "a beautiful"
  - "role model"
  - "person."
  - "human being."
- Where should the subject (alternatively, pronoun) be?
- Where should the predicate (alternatively, verb) be?
- How about the adverbs and adjectives?

- What other patterns do you see?
- Can we add some other chunks to swap in?
  - "That was"
  - "This can be"
  - "one sweet"
  - "the perfect"
  - "classmate."
  - "work of art."
- Let's choose some at random and see what we come up with!
  - "That was the perfect human being."
  - "She is one sweet classmate."

Now that your class sees where this is going, it's time for them to get in their groups to think about how this works. Give them these activity steps:

1. Create 6–10 phrases that fit the template above.
2. Cut the phrases into individual strips, then cut the strips into three appropriate chunks for mixing.
3. Label one cup "Beginnings" and put all the chunks of sentence beginnings in that cup. Do the same for "Middles" and "Ends."
4. Create an algorithm for your "compliment generator machine" (CGM) using pseudocode.
   a. What steps will your CGM need to follow in order to generate a new compliment each time you run it?
   b. How specific do you need to be to make sure your program runs without any errors?
   c. Don't forget to include steps for what to do with the papers after the compliment has been delivered.
5. Program the CGM by reading it your algorithm.
6. Run the CGM to see if it had the desired result.
7. If there was a problem (bug), fix your algorithm, then program and run the CGM again.

Step 3. Share—When students have finished their generators, allow each group to run their generator for the class. Did every CGM work the same?

Step 4. Discuss together—Gather the students together to talk about their experience.

- What was the hardest part of automating this task?
- Did you get the chance to look for issues with your algorithm and fix it for a better automation?
- Can you think of other tasks that you could create an algorithm for?
- What do you think the relationship is between algorithms and automation?

Step 5. In the real world—Algorithms are without a doubt the most important part of automating, and automation is the single largest motivation for developing computer programs. Computer scientists wouldn't even be necessary if we didn't need to automate.

More than 4,000 years ago, mathematicians were already trying to automate complex calculations using the abacus. From that point, calculators (and soon computers) became more and more intelligent. Now, they are able to automate tasks like finding unbelievably large prime numbers or sequencing the human genome—jobs that just couldn't be done by hand.

Behind every computer automation there is a computer program, and behind every computer program, there is an algorithm.