

Got Algorithms?

Created: November 2017

Last Updated: March 2018

Estimated time:	50 minutes <ul style="list-style-type: none">• [25 minutes] Activity #1• [25 minutes] Assignment
Group or individual activity:	Individual
Ages:	13-18 years old
Grades:	Grades 8-12
Online/offline elements:	This learning experience contains links to online resources, with an offline writing assignment.
Areas:	Main area: Artificial Intelligence Additional areas: Computational Thinking
License:	Creative Commons Attribution-ShareAlike 4.0 International license. For more information, please visit: http://dlrp.berkman.harvard.edu/about

Learning Goal

Understand what an algorithm is, why algorithms matter, and how algorithms are used in both everyday life and computer science.

Materials

- Paper

- Pen or pencil
- [Optional] Computer or mobile device

Resources

- Article: [What's the Deal with Algorithms?](#) - by Jacob Brogan (Slate)
- Video: [What Is an Algorithm?](#) - by Tech Policy Lab, University of Washington
- Visualization: [Sorting Algorithms in VisuAlgo](#) - by VisuAlgo

Activity #1: Algorithm Fundamentals

Introduction

Have you ever wondered how your favorite restaurant makes your favorite dish perfectly every time? Or how Siri can respond to a question you're asking? How self-driving cars work? Believe it or not, all these have one thing at their core: algorithms.

Algorithms are a set of step-by-step instructions on how to accomplish a specific task. Algorithms might be expressed in words, computer code, or even a recipe. Some algorithms may just be simple directions or processes that you use every day, but when written for a computer using a programming language, they can accomplish incredible tasks!

A Brief History of Algorithms

You've probably seen the word "algorithm" used throughout news articles about robots, computers, and artificial intelligence. While the term might sound complicated, it can be quite simple. The idea emerged long before computer science. In fact, the word originated from the name of 9th-century Arab mathematician, [Muhammad ibn Mūsā al-Khwārizmī](#) (in Latin, "Algoritmi"), known for his work on explaining step-by-step procedures that could produce a certain outcome. Algebra was also named after al-Khwārizmī!



Figure 1) A [statue of al-Khwārizmī](#) in his birthplace of Khiva, Uzbekistan

An algorithm is just a formal way to describe a set of steps needed to accomplish a goal. [You follow steps and routines all the time in everyday situations](#). For example, if you have a morning routine — getting up, making your bed, and getting ready for school or work — that’s an algorithm! In this example, the algorithm isn’t rigorously defined or written down anywhere; it’s just something you do as part of your daily routine.

Just as computers store algorithms in code on their hard drives, you store various kinds of algorithms in your brain. Now, let’s consider an everyday example that does involve formalized steps. A driver has to follow an algorithm every time they drive. For instance, when a driver comes to a stop light they follow these specific rules:



Figure 2) Photo of a [red traffic light](#)

If the light is green,
go.
If the light is yellow,
slow down and prepare to stop.
If the light is red,
stop until the light turns green again.

That's another algorithm!

Now let's explore a few algorithms for sorting numbers. Humans are perfectly capable of sorting things. Pretend that you have five cards with the following numbers on them: 3, 11, 2, 99, 12, and 23.

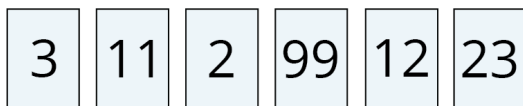


Figure 3) A set of cards

You can probably think of several different ways to take this set of cards and sort them in increasing order. Here are just two ways out of many to accomplish this task:

A) You could run through the entire stack multiple times, keeping track of the lowest number you've seen so far.

Or...

B) You could lay out all the cards and make swaps between adjacent cards so that the lower number is on the left until they're all lined up and sorted.

Each of these tactics (like "keeping track of the lowest number" or "make swaps between adjacent cards") is an algorithm. This demonstrates that multiple algorithmic approaches could accomplish the same task. However, certain solutions will result in specific benefits. For instance, [one approach may be faster or more efficient than other approaches.](#)

Algorithms in Computer Science

Since computers can execute instructions quickly, they can efficiently complete algorithmic tasks, many of which could be very difficult for humans to do. For instance,

for many people, it is difficult and time-consuming to calculate the square root of large numbers or numbers with decimals. However, using an algorithm known as the “Babylonian Method,” a computer can generate the square root almost instantaneously.

A human can also follow the Babylonian Method, which requires you to:

- 1) guess what the square root is
- 2) divide the original number by your guess
- 3) use the average of the result and your guess as your new guess
- 4) repeat until your new guess only differs from your prior guess by a very small amount

This algorithmic process makes it easier for a person to find a square root, but a computer can enter many guesses in a very short amount of time. Figure 4 illustrates the Babylonian Method written in a programming language called [Python](#).

```
def sqrt(number, difference=1e-6):  
    guess = number * 0.5  
    while abs(number - guess * guess) > difference:  
        guess = 0.5 * (guess + number / guess)  
    return guess  
  
print(sqrt(2,567.85))
```

```
>> 50.673957808720644
```

Figure 4) The Babylonian Method written in Python

This algorithm was able to calculate the square root of 2,567.85 in less than a second! That would be quite difficult for many humans to do.

Of course, AI-based technologies (like Siri and self-driving cars) do things that are significantly more complex than sorting cards or finding square roots, which is why these algorithms can be hundreds or thousands of lines of code. However, many of these algorithms also “learn,” which enable them to improve themselves without being explicitly programmed. This can lead to very advanced and exciting technologies!

Assignment

Imagine you have the opportunity to program a robot that can perform one of your daily tasks. It could be one related to work, taking care of your home, or making your favorite meal. Writing an algorithm involves listing clearly defined steps that the robot needs to follow to complete a specific goal. Your algorithm should have a minimum of five steps and maximum of ten. Write your steps down (or, type them up), specifying your goal at the top of your algorithm. Feel free to be creative with the task you'd like the robot to complete!

Example:

Algorithm Goal: Make my favorite meal for dinner — pasta with pesto!

- 1) Bring a pot of water to boil.
- 2) Combine two tablespoons of pine nuts, one cup of basil leaves, one garlic clove, and a $\frac{1}{2}$ cup extra-virgin olive oil in a blender and blend to a purée.
- 3) Add $\frac{1}{2}$ cup freshly grated Parmesan cheese and blend again.
- 4) Cook one pound of angel hair pasta in the boiling pot of water for three minutes.
- 5) Drain pasta and transfer to a large bowl.
- 6) Add the blended pesto mixture (pine nuts, basil, garlic, cheese, and olive oil) to the pasta.
- 7) Serve half of the pasta on a plate to me!