**Introduction**

**For this assignment, students are required to choose a relatively complex term from their field of study, or profession, and write three different types of definitions, varying in detail, for the chosen word. Students should write their definitions in mind that the readers do not have the technical background of the term that is to be described.**

**The goal of this assignment is to differentiate between the three types of definitions that are used and identify appropriate situations of when to use each one. Students should also understand how to adapt the definition for different intended audiences.**

The situation. Nancy Wu is a recent graduate of Computer Science and is planning to teach a computer programming summer camp for high schoolers. The children of the summer camp are all beginner programmers, specifically having never programmed before. Her task is to create a beginner computer program that the children can follow along and implement as well, but she would like to also give the students some background information about the algorithm that they will use to implement their first computer program.

Audience and use Profile. To understand what an algorithm is, readers need an overview of what it is and the common characteristics of it. It would also be helpful for readers to know about simple examples of algorithms to clarify any abstractness that a generic definition of an algorithm might present.

This audience would have beginning interest in computer science. They need an introduction to algorithms and simple examples, including a flowchart to aid in following the steps of an example.

**Definitions**

**Parenthetical Definition**: An algorithm (a sequence of actions to accomplish some task) is often illustrated by an example of a recipe.

**Sentence Definition**: An algorithm is an unambiguous sequence of instructions used to accomplish some task. Algorithms exist in everyday life, such as following a recipe, and are commonly used in technical fields, such as in computer and mathematical operations.

**Expanded Definition**:

**Origin**

An algorithm is a sequence of actions used to accomplish some task. Historically, the term algorithm is derived from the study of mathematics. The term originates from the word Algoritmi, which is the Latinization of the name Muḥammad ibn Mūsā al-Khwārizmī. He was a Persian mathematician who is also known as the Father of Algebra (Das, 2016). Algoritmi later evolved into *algorism* to refer to Latin translations of Mūsā al-Khwārizmī’s arithmetic rules (McFadden, 2017). However, with the turn of the 20th century, and the rise of the computer, algorithms have become increasingly associated with computer science, due to the nature of computer programs requiring a set of instructions to accomplish some task.

**Structure**

Many people find computer algorithms analogous to recipes. It takes “inputs” (the ingredients), performs a set of well-defined steps, and then terminates after producing an “output” (the meal) (Vencat, 2015). McFadden offers three distinguishing characteristics of an algorithm:

1. It should be finite. If the algorithm never ends, then trying to solve a problem is useless.
2. It should have well-defined instructions. Each step in the sequence must be defined with no unambiguity.
3. It should be effective. The sequence should solve what it's supposed to solve. This should also be possible to achieve by hand, with just pencil and paper.

(McFadden, 2017).

**Examples**

Computer algorithms can range from a complex set of instructions, such as Netflix’s movie recommendation system, to something simple, such as finding the largest number in a list of numbers. The inputs of finding the largest number in a list of numbers would be the list of numbers and the output would be the largest number. The sequence of steps to produce the output would be as follows:

1. Start at the first element and save the value of that element.
2. Look at the next value in the list and check if that value is larger than the one that was saved. If it is larger than the one that was saved, save that value instead and forget about the smaller value.
3. Repeat step 2 until you reach the end of the list (reaching the end of the list would be the terminating step in this algorithm).
4. The saved value will be the maximum value in the list (output).

(Kuchling, 2012).

A simple example of an everyday algorithm is presented in the flowchart in Figure 1. Figure 1 illustrates an algorithm to decide which mode of transportation one should use to get to work. Starting at “Begin” and following the directions of the arrows, the flowchart leads to a different route depending on the situation, such as owning a car, or living far from work. The algorithm certainly terminates as all different paths within the flowchart eventually arrive at “End”.



Figure 1 illustrates an algorithm of which mode of transportation to use to get to work. From Algorithm definition, 2016, https://www.c-programming-simple-steps.com/algorithm-definition.html. Copyright (2016) by c-programming-simple-steps.com

In summary, algorithms are a set of well-defined steps to perform some task. Although algorithms are commonly associated with computer science and mathematics, they are also found in everyday life, and can range from a simple set of instructions to a complex set of instructions.

**Works Cited**

Algorithm Definition. [Digital Image]. Retrieved from

https://www.c-programming-simple-steps.com/algorithm-definition.html

Das, Souvik. (May 3, 2016). The Origin and Evolution of Algorithms. Retrieved from

https://www.digit.in/science-and-technology/the-origin-of-algorithms-30045.html

Kuchling, A.M. (2012) Background: Algorithms. Retrieved from

https://study.com/academy/lesson/what-is-a-computer-algorithm-design-examples-

optimization.html

McFadden, Christopher (Jan. 30, 2017). *The Origin of Algorithms We Use Every Single Day.*

Retrieved from https://interestingengineering.com/origin-algorithms-use-every-day

Venkat, Sursesh. (Oct. 2, 2015) When an algorithm isn’t. Retrieved from

https://medium.com/@geomblog/when-an-algorithm-isn-t-\2b9fe01b9bb5