CodePlus

1. What is a computer?
   1. Understanding the basics
      1. Electronic devices
      2. Why do we use them
   2. What are algorithms?
      1. Procedure and Rules
      2. Problem solving
      3. Problems with algorithms
   3. How we use algorithms?
      1. Is this a language?
      2. Some practice

# What is a computer?

In this session we will talk about the basics because we need to understand how much you know before we can plan for complete lessons. We believe that basics are important to build a good understanding about the subject.

## Understanding the basics

Today we will talk about what makes computer a computer and how we use them, we will also learn about things that help us make computers useful for everyone. We will learn a bit about how things called algorithms help us and practise understanding them using some examples.

### Electronic devices

All computers are digital, and this means that they need electricity to work. These devices need our input to produce some output, this means that you do something with the computer, and you can see something happening, this might be as simple as moving a mouse and the cursor moves on the monitor and as complex as the processes that happen when you open an internet browser.

### Why do we use them

We use computers to help us. Anything you do in the computer is a command. You give the computer a command and computer give you some answer. Imagine that you have a book that you are reading and a helper that can help you find the page you stopped at, you don’t remember the page that you finished reading at, but you remember a few words that you read before closing the book, you tell the helper to find possible pages where you might have stopped reading. The helper then starts looking through the book to find the words you told it and finds several places which have the same words you told it. It then tells you pages which it found, and you now look at them to see if these pages are where you finished. This is simplification of how Google and other search engines work.

## What are algorithms?

Computers today are very easy to use, and we learn the easy ways to work with them. However, that wasn’t always the case. To understand how we tell computers what to do we need to understand how to make our commands for them logical and unambiguous. To do this we use logic and some basic rules to predict what we need to do.

### Procedures and Rules

Algorithm is set of commands which is done one after another to reach a goal. For example, we can think of an algorithm of making a bowl of cereal with milk. We need to remember some things that seem easy still need to be said. Assuming that you are already in the kitchen, first you need to take a bowl and place it in a convenient place on the table. So, the first step in our algorithm would be walk to the bowl, pick up the bowl, walk to the table and place the bowl on the table. We will not explore of what to pour first cereal or milk and in this example, we will first put the cereal. Second step is to walk to the cereal box, pick up the box, walk to the bowl, pour the cereal in the bowl, walk back to the place where the cereal box was, and put it down. Last step is to pour milk, so we walk to fridge, open the fridge door, take the milk, close the fridge door, walk to the bowl, open the lid, pour the milk in the bowl, close the lid, walk back to the fridge, open the fridge, put the milk in the fridge, close the fridge, and for good measure we can go back to the bowl. Now we can enjoy our delicious meal. You can see that algorithms make easy jobs long, so why we use them?

### Problem solving

Algorithms help us to solve problems in a way that gives clear instructions and in each situation will result to the same outcome. If I tell you to make a bowl of cereal you can do it in a different way, you might use different ingredients or create some other uncertainty. We think about the problems that we face by dividing them into smaller, easier problems and solve them individually. This then helps us to work collaboratively on a problem and create efficient solutions.

### Problems with algorithms

Algorithms aren’t the best solution to everything. This is because, it is easy to oversimplify the problem. For example, imagine a command “If you are too fast, slow down”. When do you need to slow down? Is there a way to know that you are going too fast, if I ask you how much is fast, you will all tell me a different thing, which is what we really want to avoid when creating algorithms, because this creates errors.

## How we use algorithms?

### Is this a language?

### Some practise