

Intro Into Programming

What is Programming?

As technology continues to take over the world and transform society, terms such as programming and algorithms are used almost constantly. But what do they mean and how are they different?

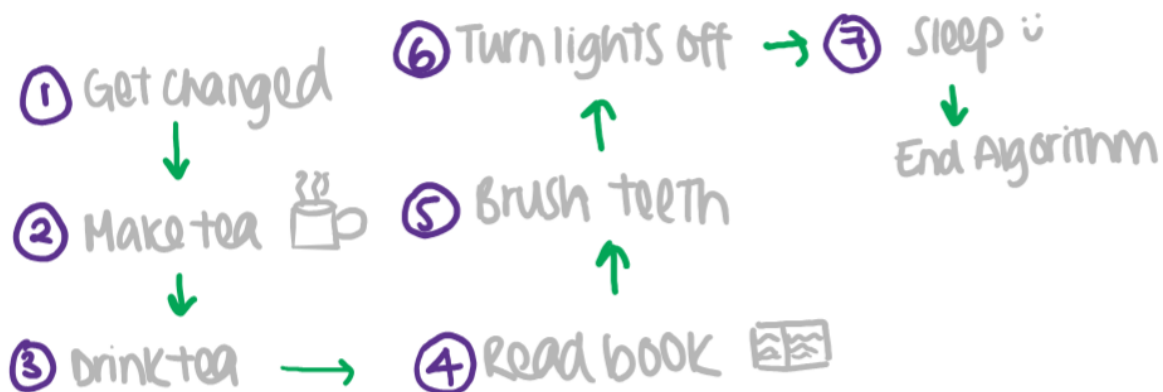
To put things simply, programming is the action and the algorithm is the object. Every programmer programs (or codes) an algorithm.

The term algorithm has become associated with highly complex, elite systems such as the social media algorithms that can seem to know exactly what you're wanting to buy next or what your favourite brand of socks are, or Artificial Intelligence that threatens to take over the world and destroy humanity...however, this is simply misunderstanding.

The definition of an algorithm is, simply, a set of step by step instructions or rules to be followed, in order to solve a problem.

The problem can be as simple as getting ready for bed.

Algorithm:



Therefore, programming is just writing the algorithm in code, so that a computer can run the algorithm and solve the problem.

Why Should You Program?

Quite simply, you should program because it's really cool. Programming is both a logic and an art, you need to be practical and creative, it's both challenging and fun. You can program by yourself and with others.

If you're a natural mathematician, it's a great extension of Mathematics. You can use mathematical concepts and the logical approach to create a physical object from your own mathematical skills.

If you're not a natural mathematician, it's a great gateway into more tricky mathematics as you'll never be scared of an unfamiliar problem!

And for everyone, with technology infiltrating nearly every part of our lives, it's neat to have a bit of insider knowledge. Being able to look at a device and understand the algorithm behind its mechanics is fulfilling and certainly helps with limiting the fear that comes with the technology revolution.

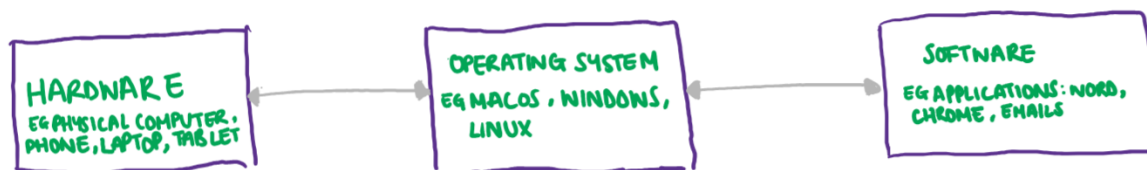
So yeah, programming is really cool and you should do it.

Also, there's an infinite demand for it and great programming skills will set you apart as an applicant for any job.

How Does Programming Work

The link between the physical hardware of the computer and all the algorithms that have been coded to make software work, is not general knowledge. The actual process of connecting the hardware to the software within any computer is achieved by the Operating System.

The role of the Operating System can be shown in this very simple diagram:



There are many complexities to the Operating System that don't need to be explained right now, however the basis is that Operating Systems are what makes the algorithms we code be able to run on a computer, which in turn allows the computer to be used.

Typically, the programs need to be translated. This can be done by a device called a Compiler or an Interpreter. Both these devices translate the code we write into binary (the language computers can understand).

Note: binary is a base 2 number system, made up entirely of 1s and 0s. It sounds really abstract, but it's actually pretty cool and can be used to achieve very complex things. It'll be explored in more depth in another resource

Different Types of Programming

As if matters weren't complex enough already, there are many different types of programming and many different languages. However, when you're starting out, don't get too bothered about the different languages.

A type of programming is, more technically, called a paradigm. The main paradigms are:

- Procedural
- Functional
- Object-Oriented

Most beginners start learning 'procedural' programming, due to it being considered easier to learn. Procedural involves writing a series of procedures (blocks of code that perform specific tasks) and executing them in a certain order.

However, the better you become at programming and you start learning the two other main paradigms, you may realise that certain paradigms are better for different programs.

Therefore, the best thing to do is to pick a multi-paradigm language. A multi-paradigm language is a language that lets you code in more than one paradigm. Examples of these are:

- JavaScript
- Python
- Java
- C++
- And many more

Conclusion

So now you should have a pretty good overview of what it means to call yourself a code and are inspired to get going!

Coding is so so much fun, and worth every second of learning it. It's a superpower.