What? Do I need to know how to program for this course?

This is a theoretical course on information theory. However, these days, even if you are studying theory, some basic programming can help you tremendously. In this course, you will be expected to write small programs for exactly that purpose: to better understand certain concepts; to develop or test hypotheses; and to observe the effects of large numbers (bigger data sets which you cannot process by hand) on information-theoretic quantities.

Do I need to be a proficient programmer to follow this course?

No. We intentionally keep the programming exercises small, a few lines of code at a time. You will never be expected to write complicated pieces of software, and the emphasis will lie on the mathematical concepts rather than algorithmic complexity.

If you know about the following programming concepts, then you are good to go:

- instantiating variables, and giving them values of basic types (boolean, integer, float, string)
- performing basic operations on those types, such as adding integers, converting strings to lowercase, etc. (or knowing where to look up how to do these things)
- printing output to the console
- if(/else) statements and for/while loops
- define and call functions
- working with some basic data structures, such as lists (arrays) and dictionaries (maps).
- if you're not using Python: read an input plaintext file, and convert it into a useful data structure. For example, you may want to create an array of all words in the file, or a list of all different symbols and their frequencies.

What if I don't know how to do these things?

There are all kinds of free tools available to learn the basics of programming. If you have no programming experience yet, we advise you to use Python, because that will allow you to use the exercise files that we provide. You can have a look at one of the following online tutorials (or find your own):

- Learn Python
- Codecademy

Additionally, you will likely have one or more **team mates** that have programming experience; they may be willing to help you get through the hard bits. You can also ask other students for help using this discussion thread.

Do you provide IT support?

No, we do not. You are free to use any language/tool you like, but you are responsible for getting that language/tool up and running. We test the Python code stubs in the online environment in which we provide them; you can always fall back on using that.

There are also plenty of online environments available which don't require you to install anything (except sometimes create an account). For example, Repl.it is what we use to provide you with the Python code stubs to get you started. You can fork and edit these stubs directly on that website. This type of website also exists for Java and C++, for example.