

# MAS115 Mathematical Investigation Skills: Python

Dr Alexander Fletcher

Autumn 2021/22

## Lecture 1: Intended learning outcomes

By the end of this lecture, you will be able to:

- explain why we use Python in this course;
- describe what a computer program is;
- explain how to calculate a factorial in Python.

## 0 Introduction and overview

### 0.1 How the Python part of the module works

- There is one lecture in Weeks 1, 3, 5 and 6.
- In Week 12 there will be an online Python test.
- There is one lab per week, on Monday afternoons.
- The lectures and labs teach different things in different ways.
- There is one homework per week (submitted/assessed online).

### 0.2 The module webpage

Visit <http://mas115.group.shef.ac.uk> regularly. There you can

- download the lecture notes;
- download the lab sheets;
- see additional material;
- find useful links.

You can also find these via the module page on Blackboard.

### 0.3 Why programming?

- It is a basic tool in the modern mathematician's toolkit.
- Programming helps you better understand algorithms; useful in your degree and the wider world.
- It is a basic skill that you will leave your degree with.
- It helps you understand how your apps do and don't work.
- It's fun! (Sometimes.)

### 0.4 Why Python?

- Python is easy.
- It is a good language for *learning* to program.
- Python is suitable for scientific programming as well as for programming web-interfaces, games, etc.
- Python is modern. Used by Google, Facebook, NASA, etc.
- Python is freely available for Windows, Mac OS X and Linux.

### 0.5 Some etymology

Python is named after "Monty Python", a British comedy group from the 60s, 70s and 80s. Tradition dictates the use of terms from their sketches in examples. The use of the word 'spam' that you are familiar with comes from that Monty Python as well.

# 1 Programming

## 1.1 What is this?

### INGREDIENTS

3 eggs, as fresh as possible  
2 knobs unsalted butter  
Salt and pepper

### INSTRUCTIONS

Crack the eggs into a small bowl and whisk.  
Add some salt and pepper.  
Heat the butter in a 9-inch non-stick frying pan.  
Pour in the eggs.  
In the first 30 seconds, create 6-10 small cuts.  
When the top is nearly set turn off the heat.  
Don't worry if some of the egg in the centre isn't quite set.  
Use your spatula to flip one half over the other.  
Serve immediately.

## 1.2 What is this?

### GLOSSARY

C12B = sl next 6 sts onto a CN and leave at back of work.  
(K1. P1) 3 times then (K1. P1) 3 times from CN.

### INSTRUCTIONS

CO 36 sts.  
1st row: (RS). \*K1. P1. Rep from \* to end of row.  
Rep last row 5 times more.  
\*\*Next row: (RS). (K1. P1)  
6 times. C12B. (K1. P1) 6 times.  
Next 5 rows: \*K1. P1. Rep from \* to end of row.\*\*  
Rep from \*\* to \*\* until work from beg measures 60 ins.  
Cast off in pat.

## 1.3 What is this?

```

1 ccaatcaggt ccctccctac ctcagatcgc agctataata cataggagta aagaggcttc
61 tgcattaag tggctgtggc ttgaagtaac gttgtgattt cgaggtcagt cttacctttc
121 gcatccccgc cgaaaccctc cgatgctgta tcagtcgcac gtttccgcac ctgtcacggt
181 cggggcttgg cgctgctgag ggacacgcgt gaaccgagga gacggcaagg acatcggcgg
241 agatccgcgc ctcgacaacg agaaaccctg ctagacagac cgctcgagaa caccgcagcg
301 agattcagcg tgcggcaaaa tgcggctttt gacgagagtg ctgctggtgt ctctctcac
361 tctgtccttg gtggtgtccg gactggcctg cggctctggc agaggctacg gcagaagaag
421 acatccgaag aagctgacac ctctcgctta caagcagttc atacctaag tcgcgagaa
481 gaccttaggg gccagcggca gatacagggg caagataacg cgcaattcgg agagatttaa
541 agaacttact ccaattata atcccacat tatctttaag gatgaggaga acacgggagc
601 ggacaggtc atgacacaga gatgcaaaga caagctgaac tcgctggcca tctctgtaat
661 gaaccactgg ccagggggtta agctgctgtg gacagagggc tgggatgagg acggtcacca
721 tttgaaaga taactccact acgaggggag agctgttgat attaccact ctgaccgaga
781 caagagcaaa tacgggacac tgtctgcct agctgtggag gctggattg actgggtcta
841 ttacgagtc aaagccaca ttcatgctc tgtcaaagca gaaaattcgg ttgctgcgaa
901 atctgggggc tgtttccag gttcggctct ggtctcgctc caggacggag gacagaaggc
961 cgtgaaggac ctgaaccccg gagacaaggt gctggcggca gacagcgcgg gaaacctggt
1021 gttcagcgac ttcacatgt tcacagaccg agactccaag acgcgacgtg tgtttacgt
1081 catagaaacg caagaaccg ttgaaagat caccctcacc gccgctcacc tccttttgt
1141 cctcgacaac tcaacgggag atctccacac catgaccgcc gcgtatgcca gcagtgtcag
1201 agccggacaa aaggtgatgg ttggtgatga tagcggccag cttaaatctg tcactgtgca
1261 gcgataatac acggaggagc agcggggctc gttcgcacca gtgactgcac atgggacct
1321 tgtggtcgac agaatactgg cgtcctgta cgcgtaata gaggaccagg ggcttgcgca

```

## 1.4 What is a computer program?

A computer program is a precise set of instructions to perform a task written in a language that a computer can understand. The two examples were precise sets of instructions written in specific languages for humans to understand.

Writing a computer program can be very roughly split into two parts:

- Analysing the task.
- Writing the code.

The first part is very mathematical and requires a careful, precise understanding of what the task requires. [In general, once you have written some code you will analyse the task again and so on.]

- The lectures will try to emphasize the first part.
- The labs will concentrate more on the second part.

## 1.5 Calculating a factorial

1. What is  $4!$ ?
2. How is  $n!$  defined? (What is  $n$ ?)
3. How do you calculate it?
4. How do you tell someone else to calculate it?
5. How do you tell a computer how to calculate it?

## 1.6 First attempt at a program

```

1 | # A program to calculate 10!
2 |
3 |
4 | factorial = 1
5 | i = 1
6 | while i <= 10:
7 |     factorial = factorial * i
8 |     i = i + 1
9 | print("10! = ", factorial)

```

Line 1 is a *comment*.

Line 4 sets the *variable* `factorial` to 1.

Line 5 sets the *variable* `i` to 1.

Line 6 starts up a *loop*. It repeats the following lines until `i > 10`.

Line 7 *sets* the variable `factorial` to `i` times its old value.

Line 8 *sets* the variable `i` to be its old value plus 1.

Line 9 *prints* out the result to the screen.

## 1.7 Adding some user input

```

1 | # A program to calculate factorial of number entered
2 | n = int(input("Enter the number you want "
3 |             "to calculate the factorial of: "))
4 | factorial = 1
5 | i = 1
6 | while i <= n:
7 |     factorial = factorial * i
8 |     i = i + 1
9 | print(n, "! = ", factorial)

```

We have added lines 2 and 3 and changed 10 to  $n$  on lines 6 and 9. (Lines 2 and 3 are one line of code but won't fit on screen.) This sets  $n$  to be the integer that the user inputs after the prompt. Note: we get the wrong answer if we enter a negative number!

## 1.8 Some terminology

What we have written is variously referred to as:

- a *program* (not “programme”),
- a *script* (often a short program), or

- *some code* (never “a code”).

Two other terms are the following:

- *App*: (or “application”) usually a program on a mobile device or website.
- *Algorithm*: process or idea behind a program (misused by journalists).

Writing programs is referred to as either:

- *programming* or
- *coding*.

These two terms mean the same thing.