

This first lab gets you to work with Python, the programming language that we will be using to write algorithms in this module. Python is similar to Java in syntax, although much simpler. In this lab you will write and run your first Python programs.

Look at `lab1.ipynb` for examples on how to use Python commands to solve the questions.

Marks (max 5): Questions 1-2: 1 mark | Questions 3-6: 1 mark each

Question 1

We will be coding Python using the Jupyter IDE. You are asked to do the following:

1. download the code file `lab1.ipynb` from QM+ and store it on your PC
2. start Jupyter Notebook by running on the command line: `jupyter-notebook`
3. open `lab1.ipynb` in Jupyter and play with (i.e. run) each of its different functions

Question 2

Write a Python function:

```
def question2(x, y, z)
```

which checks the values of the inputs `x`, `y`, `z` to see if any are equal and prints whichever one of the following applies:

1. "All three inputs have equal values"
2. "x and y have equal values"
3. "x and z have equal values"
4. "y and z have equal values"
5. "All the three inputs have different values"

The function also **returns the corresponding number** in each case. For example, running `question2(42,4,42)` in Jupyter we would get:

```
x and z have equal values
3
```

Note: you will need to use the `if-elif-else` commands, and `print`.

Question 3

Write a Python function:

```
def question3()
```

that is a modified version of `question2` which sets the three variables `x`, `y`, `z` to values typed in by the user when the code is run, and prints and returns as `question2`.

Note: this can be done using the built-in `input` function.

Question 4

Write a Python function:

```
def question4(x, y, z)
```

that is a modified version of `question2` which takes as arguments three *arrays* and tests whether their lengths are equal.

For example, if the function takes `[1,2,3]`, `[10,20,30,40]` and `[65,32,7]` as its three arguments in that order, it should print "x and z have equal values" and return 3.

Note: this can be done using the built-in `len` function which returns the length of an array.

Question 5

Write a Python function:

```
def question5(x, y, z)
```

that is a modified version of `question4` which takes three arrays as its arguments and tests whether the result of adding their contents together is equal.

For example, if the function takes `[1,2,3]`, `[10,20,30,40]` and `[65,32,3]` as its three arguments in that order, it should print "y and z have equal sums" and return 4.

Suggestion: you can use an extra function that takes an array of integers and returns the result of adding them together. For example, with argument `[65,32,3]` it would return 100.

Question 6

In Python a variable can be set to refer to a function, which also means a function can be passed as an argument to another function. Write a Python function:

```
def question6(f, x, y, z)
```

that is a modified version of `question1` which takes four arguments, one being a function and the rest being the three values being checked. The function should print whichever one of the following applies:

1. "Applying f to all three values gives the same result"
2. "Applying f to x and y gives the same result"
3. "Applying f to x and z gives the same result"
4. "Applying f to y and z gives the same result"
5. "Applying f to each of the three values gives a different result"

Observe that `question6` should work like `question4` if `len` is given as first argument. It should work like `question5` if the extra function in Question 5 is given as first argument.