# Laboratory 06 — Revising and Consolidating

# Programming Concepts

Topics covered:

* Writing functions
* Loops & Conditions
* Processing files using loops with conditions

## preparation

Lab attendance is compulsory. You will receive 1 mark for being present at the start of the lab and staying at least until the tutor has finished introducing the lab and has signed your attendance sheet.

## Exercises

The following exercises must be completed during your allocated laboratory time. You must show your work to the laboratory tutor who will sign off when the work is completed correctly.

### Exercise 6.1

[2 marks] Given an aeroplane’s acceleration a and take-off speed v, you can compute the minimum runway length needed for an aeroplane to take off using the following formula:

Write the get\_runway\_length() function which takes an acceleration a and speed v as parameters and returns the minimum runway length rounded up to the nearest integer.

**Arguments**: an acceleration a (float) and speed v (float)

**Returns**: the minimum runway length rounded up to nearest whole integer

**Extra knowledge:** Use math.ceil() to round up to nearest whole integer

**Sample output:**

|  |
| --- |
| >>> get\_runway\_length(3.5, 60.0)  515 |

### Exercise 6.2

[2 marks] Write the get\_gender() function which returns a gender description based on the code passed in as a string parameter. The gender will be determined as follows:

* If code is equal to "F", then the function should return "Female"
* Otherwise, if code is equal to "M", then the function should return "Male"
* Otherwise, the function should return "Unknown".

**Arguments**: a code (String)

**Returns**: Female/Male/Unknown

**Sample output:**

|  |
| --- |
| >>> get\_gender("F")  'Female'  >>> get\_gender("M")  'Male'  >>> get\_gender("A")  'Unknown' |

### Exercise 6.3

[3 marks] Write the five\_letter\_words() function which takes an input filename and an output filename as parameters. Your function should read the contents from the input file and write out all the 5 letter words to the output file.

**Arguments:** an input filename and an output filename

**Sample contents of input file (words.txt):**

|  |
| --- |
| The woods are lovely dark and deep  But I have promises to keep  And miles to go before I sleep  And miles to go before I sleep |

**Sample Call to function:**

|  |
| --- |
| >>> five\_letter\_words("words.txt", "five-letter-words.txt") |

**Contents of the output file (five-letter-words.txt):**

|  |
| --- |
| woods  miles  sleep  miles  sleep |

### Exercise 6.4

[2 marks] Complete the add\_mark() function which takes a filename and an additional mark as parameters. The file contains a series of marks (integer), separated by spaces. Your function should append the additional mark to the end of the file.

**Arguments**: a filename and a mark (integer) to be added to the end of the file

**Sample contents of input file (marks.txt):**

|  |
| --- |
| 70 32 67 90 80 |

**Sample Call to function:**

|  |
| --- |
| >>> add\_mark("marks.txt", 89) |

**Contents of the updated file (marks.txt):**

|  |
| --- |
| 70 32 67 90 80 89 |

## Homework Exercises

The following exercises are due at 11:30pm, Friday 18th April 2014. Include all the exercises in a single module (file), named “Lab06\_Homework.py”. Your file must include a docstring at the top of the file containing your name, UPI and ID number. Submit the file containing your exercises using the Assignment Dropbox.

### Exercise 6.5

The cumulative sum of a list [a, b, c, ...] is defined as [a, a+b, a+b+c, ...]. Write the cumulative\_sum function which takes a list of integers as input and returns the cumulative sum of the list.

**Arguments**: an integer list

**Returns**: the cumulative sum of a list.

**Sample output:**

|  |
| --- |
| >>> cumulative\_sum([1, 2, 3, 4])  [1, 3, 6, 10]  >>> cumulative\_sum([4, 3, 2, 1])  [4, 7, 9, 10] |

### Exercise 6.6

Write the unique\_file() function which takes an input filename and an output filename as parameters. Your function should read the contents from the input file and create a file of unique words.

**Arguments:** an input filename and an output filename

**Sample contents of input file:**

|  |
| --- |
| The woods are lovely dark and deep  But I have promises to keep  And miles to go before I sleep  And miles to go before I sleep |

**Sample Call to function:**

|  |
| --- |
| >>> unique\_file("words.txt", "unique\_values.txt") |

**Contents of the output file (unique\_values.txt):**

|  |
| --- |
| The  woods  are  lovely  dark  and  ... |

## Advanced Exercises (optional)

### Exercise 6.7

[0 mark] Write the generate\_random\_numbers() function that takes a filename as a parameter and writes 100 random integers to the file. The range is from 0 to 20 inclusive. Integers are to be separated by a space in the file. After that, your function should read the data back from the file and print a non-duplicate integer list.

**Arguments:** a filename

**Sample output:**

|  |
| --- |
| >>> generate\_random\_numbers("numbers.txt")  8 3 4 10 12 17 13 11 19 9 0 7 20 15 5 6 14 2 18 16 1 |

Note: your output may look different from the above example but you should have approximately 21 integers.

## ASSESSMENT

Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Lab day and time: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Check list for laboratory exercises (to be completed by Lab tutor)**

|  |  |
| --- | --- |
| On time: 🞎 (1 mark)  Exercise 6.1: 🞎 (2 marks)  Exercise 6.2: 🞎 (2 marks)  Exercise 6.3: 🞎 (3 marks) | Exercise 6.4: 🞎 (2 marks)  Teaching Assistant: \_\_\_\_\_\_\_\_\_\_\_\_\_\_ |

Total mark: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/10 Lab tutor: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

## Marking Scheme

|  |  |
| --- | --- |
| Marks | Feedback |
| 0.5 | Include a docstring at the top of the file containing your name, UPI and ID number |
| 0.5 | Include a docstring in each function. |
| 1 | Include all the exercises in a single file |
| 1 | The cumulative\_sum () function is defined correctly. |
| 3 | Well done! Your function passed test case 1, 2, and 3 |
| 1 | The unique\_file() function is defined correctly. |
| 1 | A text file is created correctly. |
| 2 | Well done! Your function passed test case 1 and 2. |