Prof. Dr. Jörg Müller
Arthur Fleig

AI VIII – Serious Games

UNIVERSITÄT BAYREUTH

Exercises for **Programming, Data Analysis, and Deep Learning in Python** (SoSe 2021)

Exercise Sheet no. 3, *Deadline*: Monday, May 3, 10:15

Notes

- Pay attention to the notes on the previous sheet.

Exercise 9 Bakery management (programming exercise) (8 points)

Suppose you are working as an intern in a bakery that produces only one kind of bread. The recipe and the current stockpile of ingredients are retrievable from the firm's internal management system as Python dictionaries. They are given as follows:

```
inventory_bakery = {'Flour': 23485, 'Salt': 329, 'Yeast': 834}
recipe_bread = {'Flour': 500, 'Water': 300, 'Salt': 9, 'Yeast': 30}
```

Hints: Find out what the operators `//` and `%` do before writing the respective functions. The bakery uses an infinite supply of tap water to bake.

a) The bakery management wants you to find out how many loafs of bread they could possibly bake today without having to send someone shopping. Write code that computes this using (one of) the above operator(s) and place the code in a function, so it is conveniently reusable. Give the function a name that is both short and descriptive. Call this function and output the result. Update the inventory accordingly using (one of) the above operator(s), assuming you have baked as many loafs of bread as was possible without shopping.

b) After baking the amount of bread loafs from a) you are now out of the ingredient that was the bottleneck. Someone has to go shopping. Your boss wants to have ingredients for another 150 loafs of bread (and not more) and it is your task to calculate the shopping "list". The list should actually be a dictionary using the ingredients as keys and the number of packages (i.e. an integer) that needs to be bought as values. The package sizes are:

```
package_sizes = {'Flour': 1000, 'Salt': 250, 'Yeast': 100}
```

Do that in a new, well-named function, which takes the amount of loafs desired and the recipe and the inventory dictionaries as arguments and returns the shopping "list". Output it.

Exercise 10 Lists, again (programming exercise) (7 points)

The management of your company has run a lottery. Unfortunately, the implementation did not prevent one and the same person from registering more than once to increase winning chances, even though the conditions of participation prohibit it. You are given a list of all participants:

```
L = ['Lucas Sankt', 'Phillipp Klug', 'Alf Redo', 'Klaudia Kaiser',
   'Ute Beich', 'Vanessa Trommler', 'Lien Hsia', 'Alf Redo', 'Alf
   Redo', 'Ute Beich']
```

Your task is to remove all duplicate entries, so that every name appears only once. The end result should be a list.

a) Write a Python script that completes the task without using the `set` datatype.

b) Write a Python script (in the same file) that completes the task using the `set` datatype.

Hints: In a), sort the list first. Then use a `for` loop to iterate over the sorted list. Use the `append` command to add elements to a new list (that has to be empty at the beginning). Use `if` to decide whether to append an element to the new list.

Exercise 11 Floats (4 points)

Try the following calculations in Python:

i) `1E308 - 1E308` *ii*) `1E308 + 1E308` *iii*) `1E308 - 2E308` *iv*) `2E308 - 2E308`

What result would you expect in each case if you calculated in your head? What does Python give you as a result in each case (and why)? (You do not need to submit your Python code.)

Hint: Consider Exercise 2.

Exercise 12 Functions (programming exercise) (5 points)

a) What happens to variables in a local scope when the function call returns?

b) Write a function `collatz(int_number)` that takes as parameter an integer `int_number`. If `int_number` is even, then the function should print and return `int_number // 2`. If `int_number` is odd, then the function should print and return `3 * int_number + 1`. Then let a user type in an integer number and store it in a variable `num`. Call the `collatz` function on `num` and save the result in the variable `num`. Keep doing this until the `collatz` function returns the value 1.
Hint: Use a `while` loop for the second part.