Computational Thinking and Programming – A.Y. 2022/2023

Written examination – 16/03/2023

Given name: _____

Family name: _____

Matriculation number: _____

University e-mail: _____

Enrolment a. year: [ ] 2022/2023   [ ] 2021/2022   [ ] 2020/2021   [ ] 2019/2020   [ ] other

Is it your first try?   Yes | No

The examination is organised in three different sections:

- Section 1: basic questions [max. score: 16]. It contains four simple questions about the topics of the whole course. Each question requires a short answer. Each question answered correctly will give you 4 points (or less for partial answers).

- Section 2: understanding [max. score 8]. It contains an algorithm in Python, and you have to report the particular results of some of its executions according to specific input values.

- Section 3: development [max. score 8] It describes a particular computational problem to solve, and you are asked to write an algorithm in Python for addressing it.

You have 1 hour and 30 minutes for completing the examination. By the final deadline, you should deliver only the original text (i.e. this document) with the definitive answers to the various exercises that must to be written with a pen – pencils are not permitted. You can keep all the draft papers that you may use during the examination for your convenience – blank sheets will be provided to you on request.

Section 1: basic questions

1 – Being the variable *test* a set, which of the following Python codes are valid instructions?

- `test.append(1)`
- `test.add(1)`
- `test.update({1, 2, 3})`
- `del test["a"]`
- `test["a"] = 1`
- `test["a"]`

2 – Consider the following Python function:

```
def f(s):
    res = 0
    for i in range(len(s)):
        res = res + i
    return res
```

What is the value returned by `f("hive")`?

3 – Write down a small function in Python that takes in input a string returns None if it is empty, otherwise it returns the number of non-vowel characters it contains (e.g. it returns 3 if the input is "Hello").

4 – Write down the name and describe the main steps of the fibonacci recursive algorithm.

Section 2: understanding

Consider the following functions written in Python:

```python
def cnt(mat_string):
    result = 0
    if len(mat_string) > 0:
        n = int(mat_string[0])

        if n % 2 == 0:
            return 1 + cnt(mat_string[1:])
        else:
            return -1 + cnt(mat_string[1:])

    return result
```

Consider the variable `my_mat_string` containing string of your matriculation number. What is the value returned by calling the function `cnt` as shown as follows:

```python
cnt(my_mat_string)
```

Section 3: development

The **index of coincidence** (IC) provides a measure of how likely it is to draw two matching letters by randomly selecting two letters from a given text. The chance of drawing a given letter in the text is calculated by number of times that letter appears divided by length of the text (excluding spaces and punctuation, and considering all letters in lowercase). The chance of drawing that same letter again (without replacement) is the number of occurrences of that letter minus one, divided by the lengh of the text minus one. The product of these two values gives you the chance of drawing that letter twice in a row. One can find this product for each letter that appears in the text, then sum these products to get a chance of drawing two of a kind. This probability is then be normalized by multiplying it by some coefficient c dependant of the language of the text, as shown in the following formula:

$$IC = c * \left(\left(\frac{n_a}{N} * \frac{n_a - 1}{N - 1}\right) + \left(\frac{n_b}{N} * \frac{n_b - 1}{N - 1}\right) + ... + \left(\frac{n_z}{N} * \frac{n_z - 1}{N - 1}\right)\right)$$

where n_a is the number of occurrences of the letter a in the text, n_b is is the number of occurrences of the letter b in the text, and so on (considering all letters in the alphabeth), and N is the total number of letters in the text.

Write an algorithm in Python – `def ic(s, c)` – which takes in input a string `s` representing a text and a number `c` representing the coefficient mentioned in the formula aboce, and returns a number representing the index of coincidence for the input text.