

Written examination – 25/05/2023

Is it your first try?	Yes	No
-----------------------	-----	----

- 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 string, which of the following Python codes are valid instructions?

- `test.append("d")`
- `test.add("f")`
- `test.split("abc")`
- `del test["a"]`
- `test["g"] = 1`
- `test[1]`

2 – Consider the following Python function:

```
def f(i1, i2):  
    res = 0  
    for I in range((i1 + i2) % 2):  
        res = res + 1  
    return res
```

What is the value returned by `f(554, 36)`?

3 – Write down a small function in Python that takes in input a boolean and an integer and, if the boolean is True, it sums one to the input integer and returns it, otherwise it subtract one to the input integer and then returns it.

4 – Explain the main passages of the merge sort algorithm.

Section 2: understanding

Consider the following functions written in Python:

```
def do(five_chars):
    idx = 0

    alphabeth = list("abcdefghijklmnopqrstuvwxyz")
    for c in five_chars:
        if c in alphabeth:
            idx += alphabeth.index(c)
        else:
            idx -= 1

    result = set()
    idx = idx % 5
    for i in range(idx):
        result.add(five_chars[i])

    return result
```

Consider the variable `my_five_chars` containing string of your first five characters of your surname. What is the value returned by calling the function `do` as shown as follows:

```
do(my_five_chars)
```

Section 3: development

The **Vigenère cipher** is a method of encrypting alphabetic text where each letter of the input text is replaced by a letter some fixed number of positions down the alphabet, and the number of positions is determined by the corresponding letter of another input text, the *key*. For example, if the input text is "another exam" and the key is "bucainangolo", then:

- the first letter *a* of the input text is shifted by 1 position in the alphabet (because the first letter *B* of the key is the 2nd letter of the English alphabet, counting from 0), yielding *b*;
- the second letter *n* is shifted by 20 (because the second letter *U* of the key means 20) yielding *h*, with wrap-around;
- the third letter *o* is shifted by 2 (*C*) yielding *q*, with wrap-around;
- and so on; yielding the message "bhqtprr klla" (all spaces are preserved).

Write an algorithm in Python – `def vigenere(text, key)` – which considers only English texts, and takes in input a string `text` in lowercase representing the input text to cipher and another lowercase string `key` representing the key for the cipher – where both `text` and `key` contain the same number of characters, i.e. `len(text)` is equal to `len(key)`. The algorithm must return the encrypted text according to the rules described above.