

Exercise 4 – Submission date – 30.5

In this exercise you are allowed to import NumPy, SciPy and Matplotlib.

Each part should be submitted in a separate code file (ex4a.py, ex4b.py) in addition to the answer file (pdf). In this exercise you are not asked to check for the validity of the parameters given to the functions.

Part A – getting familiar with NumPy:

In this part you should use NumPy's function and methods. You are not allowed to change the NumPy object to a list object.

1. Write a function that finds the indices of the N largest numbers within an array.
The function receives a NumPy's array which contains numbers and additional number N .
The function returns the largest N number from the list given to the function.
Testing example:

```
arr = numpy.array([1,2,3,5,1,3,4])  
find_largest_n(arr,2) == numpy.array([4,5])
```
2. Write a function that finds even numbers that are larger by a given number.
The function receives a NumPy's array which contains numbers and additional number.
The function returns a Numpy's array which include the even numbers that are larger than the given number.
Testing example:

```
arr = numpy.array([1,2,3,4,5,6])  
find_even_larger_than(arr,4) == numpy.array([6]) - note that 4 is not there.
```
3. Write a function that adds a prefix for each string in the array.
The function receives a NumPy's array which contains strings and additional string.
The function returns a NumPy's array with the string after adding the additional prefix.

```
arr_str = np.array(['a','b','c']), st = 'd'  
add_prefix(arr_str, st) == ["da","db","dc"]
```
4. Write a function that counts the number of strings containing a substring.
The function receives a NumPy's array which contains strings and an additional string.
The function returns the number of strings that contains the given substring.

```
arr_str = numpy.array(["Count","me","please"]), st = "e"  
count_string_with_substring(arr_str, st) == 2
```

Part B – Editing images:

1. In this question you will implement a class *MyImage* which includes methods which are designed for editing images:
 - a. Constructor – receives only a two-dimensional array of pixels (NumPy's array). Each item in the array includes another NumPy's array with 3 items (RGB, range 0-255).
 - b. *turn_area_black(self, upper_left, bottom_right)* –
The method receives two points. Each point is a list of two numbers (int) representing a point in the image – [x,y] . The method changes the image area limited by those points to black.
The function **returns** the image after the modification.
 - c. *adding_cage(self, upper_left, bottom_right, cage_color)* –
The method receives two points as mentioned in the previous method ([x,y]). In addition, the method receives a *cage_color*, an array that contains 3 values representing the color of the cage (RGB, range 0-255).
The method changes the image by adding a cage to the image.
What do you mean by cage? A cage is a rectangle with vertical lines.
You are allowed to choose how dense are the vertical lines within the rectangle.
The method **returns** the image after adding the cage.
 - d. *blur_certain_area(self, upper_left, bottom_right)* –
The arguments to this method are similar to the first method.
The method blurs the area created by the two points only.
The function **returns** an image after blurring the chosen part.
(use google to find how you can blur an image, you are allowed to use any filter)

You can use matplotlib.image to read images from your computer in order to try your class by yourselves.

2. Explain how you implemented the methods and add examples for using the methods on two images. In other words, add to the answer file images before and after applying the methods from questions 1.
While implementing these methods, pay attention for the common stages across methods.
The students that will create the most amusing images, using the methods, will earn 3 bonus points.
3. In addition to the class, you will implement a function named *image_hist(img)*. The function plots an [histogram](#) that present the distribution of each color in the image (red, green and blue).
The function receives an image (same type as in question 1) and present the image's histogram as defined above. The function does not return anything.
Run the function on an image before and after applying one of the methods from question 1. Attach the figures to the answer file. Explain in the answer file how does the manipulation influence the histograms.
Please make sure that your figures can be understood easily (add to your plot: legend, title, axis title...).