

IT1007 Introduction to Programming with Python and C

Lab Exercise 05

Submission instructions [Coursemology]:

1. There are two parts in this lab. For Part A, you have to submit within the same day of your lab session. For your Part B, you have **six days** to work on it. (E.g. if your TLab is on Monday, then your deadline is the coming Sunday midnight.)
2. Complete your code using the skeleton files provided, then **test your code on your computer first** before submitting to Coursemology.
3. To submit your code on Coursemology, click on “Labs” in the sidebar followed by the appropriate “Attempt” button.
4. **Copy ONLY the required function** from your completed skeleton file into the Coursemology code window.
5. Click “Run Code” to test that your function works on Coursemology.
6. Click “Finalise Submission” to submit your code for the **ENTIRE Part A/B. You will not be able to amend your code after you have finalised your submission.**
7. You must name your functions exactly as the questions state.

Failure to follow each of the instruction will result in 10% deduction of your marks.

Important Note: In this exercise, you should manipulate the images with Numpy and Scipy packages ONLY, but not other library such as PIL. And the package matplotlib should be used for showing the image only, namely only `imshow()` and `show()`.

Part A (Deadline: Same day of your TLab)

Question 0: Prepare an Image (0 mark)

Before you start, please prepare an image of your favorite. You can choose a picture you like. However, because of the server capacity and grading purposes, here are some limitations you need to observe. Your image should be

1. Colored with some contents
2. Min dimensions: (300 x 200 pixels), Max dimensions: (1000 x 800 pixels)
3. File size < 1MB. (Better to save your picture in format like jpg in order to save space)
4. Not containing any “bad” contents like obscene, sexual, violent, racist or any controversial contents

So I use this image as an example:

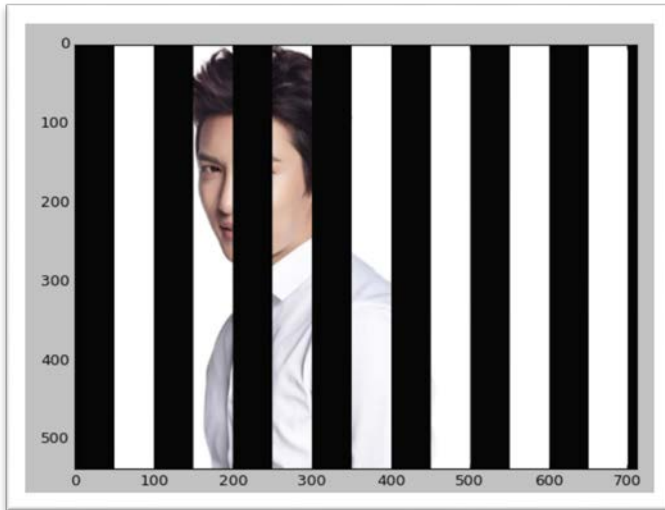


Question 1 (5 marks)

Write a function `show_an_image(filename)` that will read in an image with the `filename` and show it on the screen.

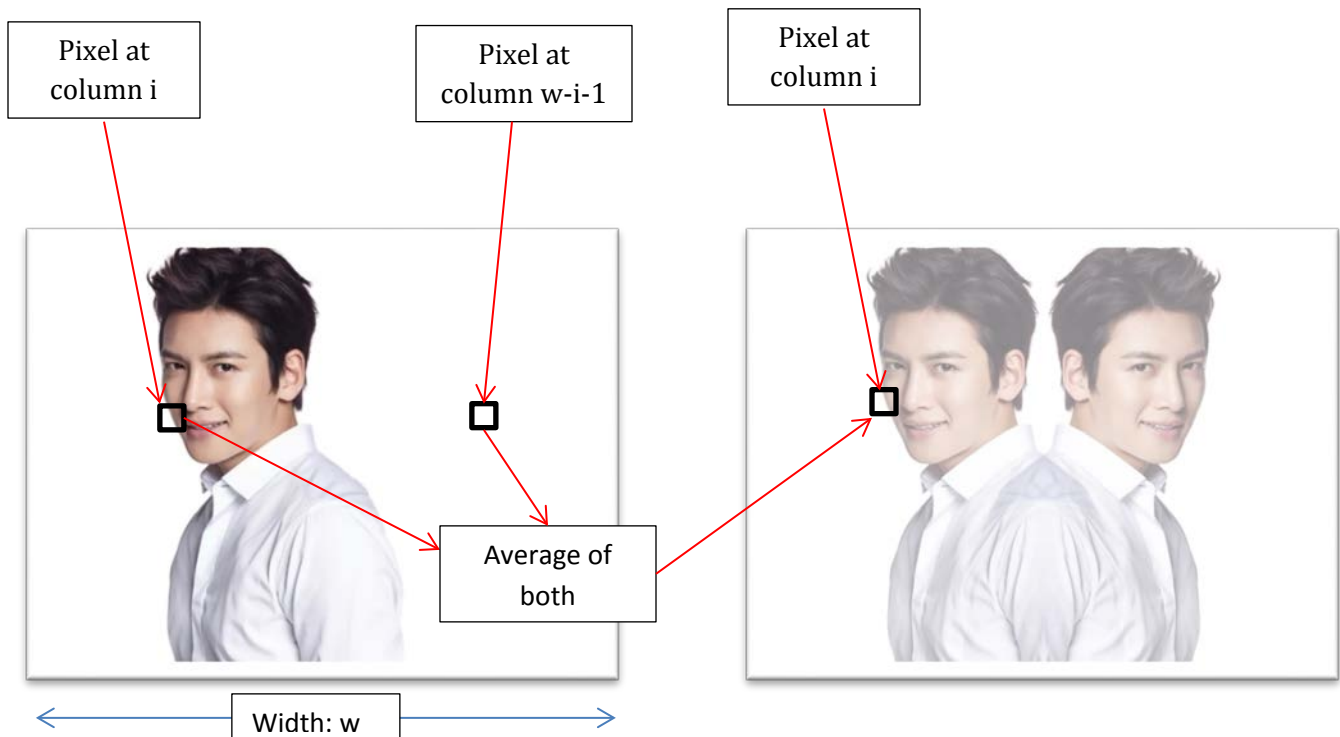
Question 2 (20 marks)

Write a function `put_behind_bar(filename)` that will read in an image and put some vertical bars with width and spacing equal to 50 pixels. The color of the black bars are (0,0,0). And this function will save the result image as `put_behind_bar_output.jpg`. Note that the leftmost column is the black bar.



Question 3 (20 marks)

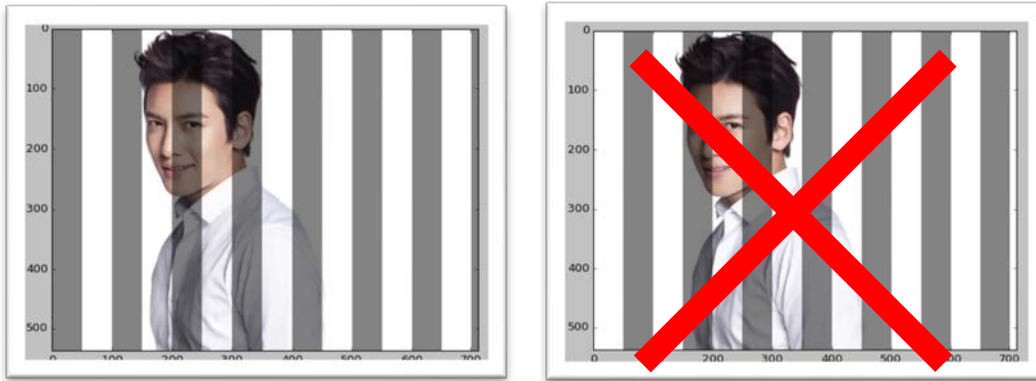
Write a function `mirror_image(filename)` that will read in an image, flip the image horizontally and overlay the flipped and original pictures together. Each pixel of the final picture will be the average of the pixels of the original and flipped images. And this function will save the result image as `mirror_image_output.jpg`.



Part B (Deadline: +6 days)

Question 4 (15 marks)

Similar to Question 2, write a function `put_behind_bar_transparent(filename)` that will read in an image and put some vertical bars with width and spacing equal to 50 pixels. The color values "in the bar" are "half" of the values of the original pixel. And this function will save the result image as `"put_behind_bar_transparent_output.jpg"`. Same as Question 2, your leftmost bar is transparent.

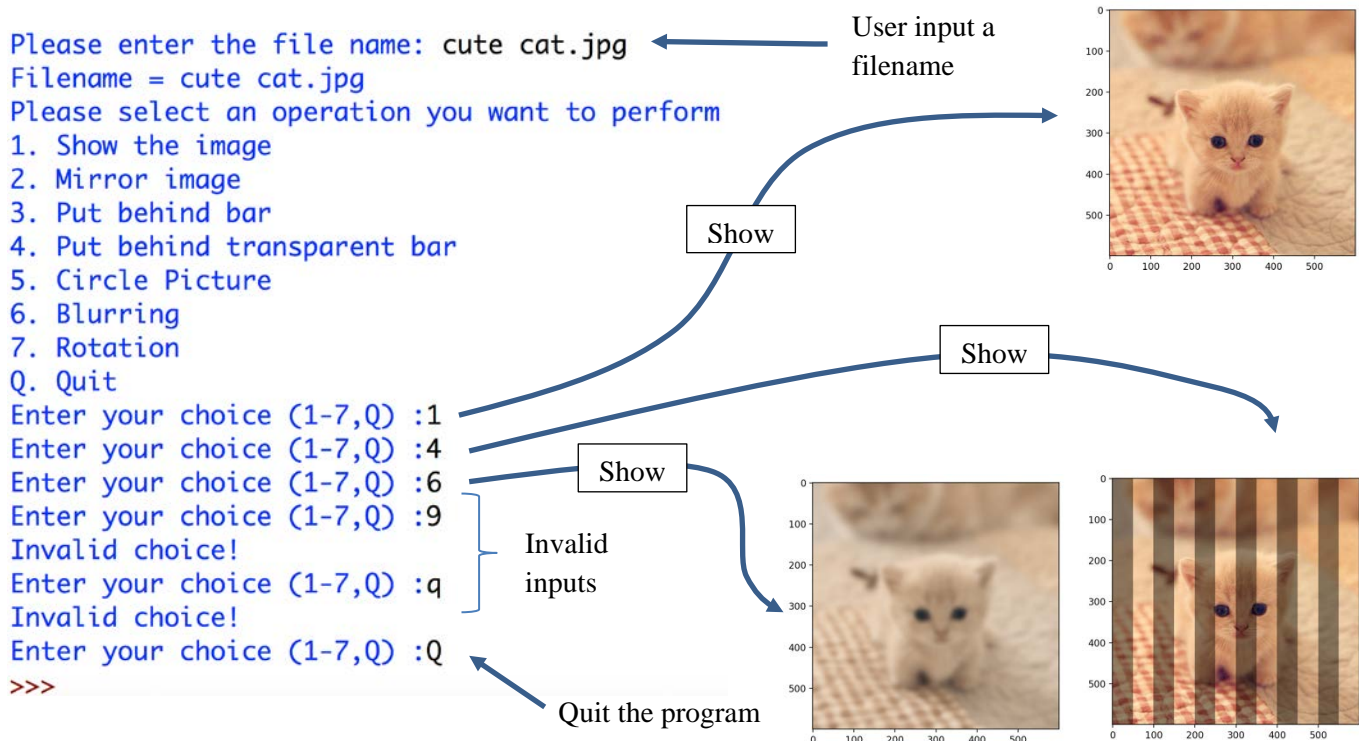


Question 5 Image Processor (40 marks)

You are going to write an image processor! For this question, you have to follow the instructions (tasks) and implement it with "a dictionary of functions". But before that, let's see how it should work. When it starts, it will ask for a file name:

```
>>> image_processor()  
Welcome to IT1007 Image Processor!  
Please enter the file name:
```

Then, depends on which function you choose, it will perform the operation of your choice:



Task 1(5 marks)

Import the ready-made functions **circle_pic**, **blur_image** and **rotate_image** from the files "**circle_pic.py**", "**gauss_filter.py**" and "**rotate_image.py**" respectively. The import of the first function is done for you as an example in the template already.

Task 2(10 marks)

Establish a dictionary of functions. The key is the command strings of '1' to '7' and the values are the functions. The one for **circle_pic** is done for you already as an example.

Task 3(5 marks)

Print a statement and ask for the filename of the image. And give a feedback of "Filename = ...".

```
>>> image_processor()  
Welcome to IT1007 Image Processor!  
Please enter the file name: Ji_Chang_Wook.jpg  
Filename = Ji_Chang_Wook.jpg
```

Task 4(5 marks)

Print the menu as shown in the previous page.

Task 5(15 marks)

Complete the image processor following the flow chart below. It should work for all the functions listed in the menu in the previous page.

