CSCI1530 Computer Principals and Java Programming 2018-2019 Second Term
Department of Computer Science and Engineering
The Chinese University of Hong Kong

Due date: 9 April 2019 (Tue) **Assignment 5** Full mark: 100
Expected normal time spent: 5 hours

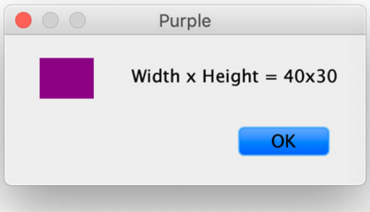
Photo Kiosk

Aims: 1. Implementing an interesting photo processing app.
2. Practising object-oriented programming.
3. Reading PPM image files.
4. Declaring, creating and using 2D arrays for image representation and processing.

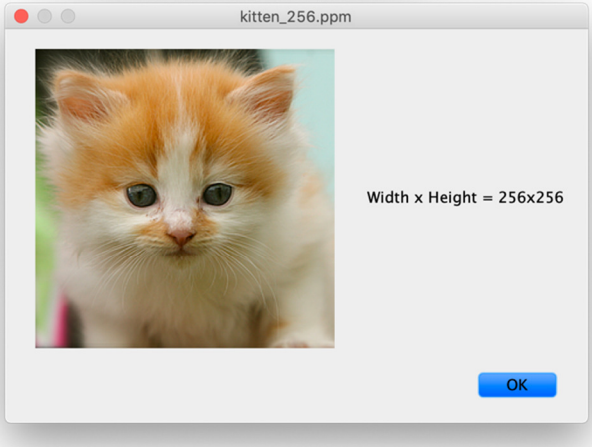
Requirements:

1. We are going to implement an interesting photo processing app for PPM[1] images.

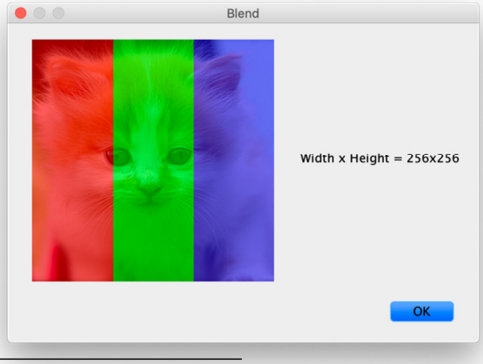
2. Firstly, the program prepares and displays some simple "default" images for your reference.



Then, it reads and shows some given PPM files:



Lastly, it blends two images together:



Kitten Figure: adopted from public domain image source - https://all-free-download.com/free-photos/download/newborn-kitten_514764.html

and writes resultant image to "blend.ppm".

[1] Portable Pixel Map (PPM) is a simple image file which supports RGB full colour images in ASCII text format.
Reference: https://en.wikipedia.org/wiki/Netpbm_format#PPM_example

3. The program may display some text via System.out for your own debugging purposes.

4. A client class PhotoKiosk is given. You are expected to complete the implementation of another class PPM.

5. When reading a PPM image file, there may be various kinds of errors and issues such as file not found, wrong file format, etc. Follow the given sample code to throw an Exception to indicate header problem. You need NOT consider other kinds of file format issues. You may keep the given Exception handling (try-catch) blocks.

6. Here is the expected standard PPM file format, *free from additional features* such as comment:

```
P3
width height
max_value
R G B R G B R G B...
R G B R G B R G B...
...
R G B R G B R G B...
```

The first two characters on the first line must be "P3".

Then 3 numbers follow: *width* and *height* define the size of the image;
max_value is usually 255 which indicates maximum possible value of a pixel component value.

R, G, B's are red, green and blue pixel component values defining a full colour image in RGB. All values shall fall within 0 and *max_value*, i.e., usually within 0 – 255.

There are *height* pixel lines run from top to bottom row-by-row. Each pixel line contains *width* number of pixels, well, R, G, B triplets.

All numbers are delimited by white-spaces (including space, tab and newline.)

Therefore, Java Scanner class is well suited for reading the PPM file in plain text ASCII format.

7. The *origin* of the image and window coordinates system is always at the top-left corner.

Your Task:

1. Before you code, revise the concepts in OOP, specifically how to define methods and constructors. You will deal with 2D array of Color in this exercise.

2. **Create** a new NetBeans project named **`PhotoKiosk`**, with a package named **`photokiosk`** and main class **`PhotoKiosk`**. Copy the given code, Java source files and PPM files from Blackboard into the proper locations under your project folder.

3. The main class should include a proper *header comment block*, similar to the one appeared in assignment 1. It should include course code and course name, title of the assignment, brief description of your work, your name, your SID, date of the work, as well as your statement of originality and declaration of understanding the guideline on academic honesty.

4. Zip and Submit your *whole NetBeans project folder* in an archive file **`PhotoKiosk.zip`** via our Online Assignment Collection Box on Blackboard **`https://blackboard.cuhk.edu.hk`**.

Marking Scheme and Notes:

1. The submitted program should be free of any typing mistakes, compilation errors and warnings.

2. Comment/remark, indentation, style are under assessment in every programming assignments unless specified otherwise. Variable naming, proper indentation for code blocks and adequate comments are important.

3. Remember to "Submit" before the cutting line. No late submission would be accepted.

4. If you submit multiple times, **ONLY** the content and time-stamp of the **latest** one would be counted. You may delete (i.e. take back) your attached file and re-submit. We ONLY take into account the last submission.

University Guideline for Plagiarism

Attention is drawn to University policy and regulations on honesty in academic work, and to the disciplinary guidelines and procedures applicable to breaches of such policy and regulations. Details may be found at http://www.cuhk.edu.hk/policy/academichonesty/. With each assignment, students are required to submit a statement that they are aware of these policies, regulations, guidelines and procedures.

Faculty of Engineering Guidelines to Academic Honesty

MUST read: http://www.erg.cuhk.edu.hk/upload/ENGG_Discipline.pdf