Features:

Exposed in UI and command-line

- Load
- Upload Script
- Save
- Blur
- Red-component
- Green-component
- Blue-component
- Luma-component
- Sepia
- Horizontal-flip
- Vertical-flip
- Color-Correct
- Compress
- Levels-Adjust
- Sharpen
- Resize Image

Exposed only in UI

- Resize Image (Image Downscaling)

Exposed only in Command Line

- Intensity-component
- Value-component
- RGB Split
- RGB Combine
- Brighten
- Histogram

Support Split

- Blur
- Sharpen
- Sepia
- Red-component
- Green-component
- Blue-component
- Luma-component
- Intensity-component
- Value-component
- Color-correct
- Levels-Adjust

How to run the code:

1. First, install IntelliJ or a similar IDE.
2. Download the zip file, unzip it, and keep it in the desired folder.
3. Make sure you have the src, res, test, out, and docs directories.
4. Expand the src directory and open Main.java.
5. Follow the below steps to either run using GUI or run using JAR.

To run using IDE -

- Run the main.java class, this will automatically open the Graphical User Interface.
- First, load the desired image by choosing the file from the system by selecting the load button and type the image name. Initially, only the load button and the Upload Script button will be visible. Only after the image is successfully loaded the other manipulation buttons will be visible.
- Make desired manipulation to the image by clicking the button for other operations which will now be available.
- Save the final image.

To run using JAR (3 ways) -

- Open command prompt/terminal and navigate to the respective folder:
    1. Type `java -jar <Program.jar>` and press Enter.
        - This will now automatically open the GUI. Now, load the desired image by choosing the file from the system by selecting the load button and type the image name. Initially, only the “Load” and the “Upload Script” buttons will be visible. You can also click on the “Upload Script” and upload the script file which contains multiple commands which are to be executed. Only after the image is successfully loaded the other buttons to manipulate the image can be used.
        - Make desired manipulation to the image by clicking the button for other operations which will now be available.
        - Save the final image.

    2. Type `java -jar Program.jar -file path-of-script-file` and press Enter.
        - This will now automatically run all the commands in the script file.
        - If any error occurs while executing any command, a message starting with “Error: “ will be displayed in the terminal.
    3. Type `java -jar Program.jar -text` and press Enter.
        - This will now ask for input from the user.
        - Follow the rules below to use the available operations.

Rules for all commands:

1. load image-path image-name: Load an image from the specified path and refer it to henceforth in the program by the given image name.
2. save image-path image-name: Save the image with the given name to the specified path which should include the name of the file.
3. red-component image-name dest-image-name: Create an image with the red-component of the image with the given name, and refer to it henceforth in the program by the given destination name. Similar commands for green, blue, value, luma, and intensity components should be supported. Note that the images for value, luma, and intensity will be greyscale images.
4. horizontal-flip image-name dest-image-name: Flip an image horizontally to create a new image, referred to henceforth by the given destination name.
5. vertical-flip image-name dest-image-name: Flip an image vertically to create a new image, referred to henceforth by the given destination name.
6. brighten increment image-name dest-image-name: brighten the image by the given increment to create a new image, referred to henceforth by the given destination name. The increment may be positive (brightening) or negative (darkening).
7. rgb-split image-name dest-image-name-red dest-image-name-green dest-image-name-blue: split the given image into three images containing its red, green, and blue components respectively. These would be the same images that would be individually produced with the red-component, green-component, and blue-component commands.
8. rgb-combine image-name red-image green-image blue-image: Combine the three images that are individually red, green, and blue into a single image that gets its red, green, and blue components from the three images respectively.
9. blur image-name dest-image-name: blur the given image and store the result in another image with the given name.
10. sharpen image-name dest-image-name: sharpen the given image and store the result in another image with the given name.
11. sepia image-name dest-image-name: produce a sepia-toned version of the given image and store the result in another image with the given name.
12. compress percentage image-name dest-image-name: modifies the pixel values of the specified image by a given percentage, resulting in a compressed version of the image (in terms of size), which is then stored with a new name.
13. color-correct image-name dest-image-name: color corrects an image by aligning the meaningful peaks of its histogram.
14. histogram image-name dest-image-name: produces an image that gives a histogram for the given image.

15. levels-adjust b m w image-name dest-image-name: adjusts the color meaningfully in an image where b, m, and w are the three relevant black, mid, and white values respectively. The values should be in ascending order.
16. Operations Preview: For certain operations, a split view of the image can be seen. The operated image is on the left and the original image is on the right. blur, sharpen, sepia, greyscale (red component, green component, blue component, luma component, value component, intensity component), color correction, and levels adjustment. To use this feature, give the command as mentioned earlier followed by "split percentage_value".

Order on how the above commands should be called:

1. Load the given image with the command: "load image-path image-name". The loading operation should have the absolute file path of the image. The image path should have the file type at the end.
2. Call the desired operation and make sure you follow the rules as mentioned above. For levels-adjust operation make sure the black, mid, and white values are in ascending order. The values should be within the range of 0 to 255. We support split preview operation only for blur, sharpen, sepia, greyscale (red component, green component, blue component, luma component, value component, intensity component), color correction, and levels adjustment.
3. After the desired operation is performed save the image with the command: "save image-path image-name". The file path should be the absolute file path of the image. The image path should have the file type at the end.