Task 1 - Application for image processing and analysis, elementary operations, noise removal.

This excercise is composed of four elements:

- Create a command line application for image processing and analysis which will be used in this and all the next excercises. The application should be able to load the given images, process them and save the results depending on the command line arguments. The application should use the following command line format:

  ```
  name --command [-argument=value [...]]
  ```

  When the `--help` command is specified the application should print all the available commands with detailed description of their arguments. To load and save images the existing libraries can be used. This part of the excercise is common for all the groups.

- Implement some elementary operations on images enabling to easily modify brightness, contrast and RGB components as well as performing some simple geometric operations. This part of the excercise is common for all the groups. The elementary operations:

  - (B1) Image brightness modification (`--brightness`).
  - (B2) Image contrast modification (`--contrast`).
  - (B3) Negative (`--negative`).

  Geometric operations:

  - (G1) Horizontal flip (`--hflip`).
  - (G2) Vertical flip (`--vflip`)
  - (G3) Diagonal flip (`--dflip`).
  - (G4) Image shrinking (`--shrink`).
  - (G5) Image enlargement (`--enlarge`).

- Implement the methods of image noise removal (from the assigned excercise variant). Compare their effects and, additionally, analyze their effects on images with various nosie models. The available variants are:

  - (N1) Median filter (`--median`), geometric mean filter (`--gmean`).
  - (N2) Min filter (`--min`), max filter (`--max`), median filter (`--median`).
  - (N3) Alpha-trimmed mean filter (`--alpha`), contraharmonic mean filter (`--cmean`).
  - (N4) Midpoint filter (`--mid`), arithmetic mean filter (`--amean`).
  - (N5) Adaptive median filter (`--adaptive`), min filter (`--min`), max filter (`--max`).
  - (N6) Median filter (`--median`), harmonic mean filter (`--hmean`).
  - (N7) Alpha-trimmed mean filter (`--alpha`), geometric mean filter (`--gmean`).
  - (N8) Midpoint filter (`--mid`), arithmetic mean filter (`--amean`).
  - (N9) Adaptive median filter (`--adaptive`), arithmetic mean filter (`--amean`).

  The analysis of the obtained results should be performed with the following similarity measures of the initial and resulting images (the comparison should be done with respect to the original image without noise!):

  - (E1) Mean square error (`--mse`).
  - (E2) Peak mean square error (`--pmse`).
  - (E3) Signal to noise ratio (`--snr`).
  - (E4) Peak signal to noise ratio (`--psnr`).
  - (E5) Maximum difference (`--md`)

In all the elements of the excercise the following issues will be taken into account:

- The code organization, method of internal data representation (data structures), generality and usefulness of the proposed solution. The report should contain the technical description of the project. The approximate weight of this part of the excercise is 0.3.
- Functionality and correctness of the implemented elementary operations and understanding of their principles. The approximate weight of this part of the excercise is 0.2.
- Performance of noise removal methods. The report should contain conclusions from the experiments and evaluation of the effectiveness of the implemented methods. The basic experiments which should be performed include: comparison between methods from the assigned excercise variant and analysis of their results for various types of noise (which of them and in what situations are better). Special attention should be also paid to possible additional "side" effects of the filters (e.g. how do they influence thin lines, separated points, etc.) and these observations should be also included in the report. The approximate weight of this part of the excercise is 0.5.

The report template:

- **report_1.doc** - Microsoft Word

Instructions and remarks:

- Elementary image operations and geometric transforms

  Elementary operations:

  - Image brightness modification. In order to decrease the brightness of an image a constant should be subtracted from individual pixels. It should be noted that the result of such an operation may not fall out of the range of valid pixel values. Per analogiam, if we want to increase brightness, a constant should be added to individual pixels. This description is valid for 24-bit images. For paletted images the addition/subtraction is performed only for individual palette entries.
  - Contrast modification. This operation is performed with exponential or linear functions.
  - Image negative. This method requires performing negation of the RGB components for palette entries or for individual image pixels (dependant on the image format).

  Simple geometric transforms:

  - Horizontal flip, vertical flip and diagonal flip. These operations are illustrated in the following pictures:

Horizontal flip

Vertical flip

Diagonal flip

- Noise removal

  The examples of image noise removal methods are presented below. In all the formulas *Sxy* denotes rectangular window around the point *(x, y)* of width *m* and height *n* (*f* denotes the actual image and *f* with a hat the processed one):

  - Median filter

    $$\hat{f}(x,y) = \underset{(s,t)\in S_{xy}}{median}\{f(s,t)\}$$

    In order to find the median (according to the definition) the elements are ordered in a sequence, sorted and the middle element is selected.

  - Min filter

    $$\hat{f}(x,y) = \min_{(s,t)\in S_{xy}}\{f(s,t)\}$$

  - Max filter

    $$\hat{f}(x,y) = \max_{(s,t)\in S_{xy}}\{f(s,t)\}$$

  - Midpoint filter

    $$\hat{f}(x,y) = \frac{1}{2}\left[\min_{(s,t)\in S_{xy}}\{f(s,t)\} + \max_{(s,t)\in S_{xy}}\{f(s,t)\}\right]$$

- Alpha-trimmed filter

  $$\hat{f}(x,y) = \frac{1}{mn-d} \sum_{(s,t)\in S_{xy}} f(s,t)$$

  where *d<mn* means that it is assumed that the mean is computed only for those pixels which remain after rejecting *d/2* greatest and *d/2* smallest values in the window *(x, y)* (naturally, in this case the summing is defined only for those *mn-d* pixels which remained).

- Arithmetic mean filter

  $$\hat{f}(x,y) = \frac{1}{mn} \sum_{(s,t)\in S_{xy}} f(s,t)$$

- Geometric mean filter

  $$\hat{f}(x,y) = \left[\prod_{(s,t)\in S_{xy}} f(s,t) \right]^{\frac{1}{mn}}$$

- Harmonic mean filter

  $$\hat{f}(x,y) = \frac{mn}{\sum_{(s,t)\in S_{xy}} \frac{1}{f(s,t)}}$$

- Contraharmonic mean filter

  $$\hat{f}(x,y) = \frac{\sum_{(s,t)\in S_{xy}} f(s,t)^{Q+1}}{\sum_{(s,t)\in S_{xy}} f(s,t)^{Q}}$$

  where Q is a real number called the *order* of the filter

- Adaptive median filter. This filter is described by the following algorithm:

  Notation:
  ```
  zmin - minimum intensity value in the Sxy window
  zmax - maximum intensity value in the Sxy window
  zmed - median of intensity values in the Sxy window
  zxy - intensity value at coordinates (x, y)
  Smax - maximum allowed size of Sxy
  ```
  Algorithm:

  Stage A:
  ```
  1. Compute A1=zmed-zmin, A2=zmed-zmax
  2. If A1>0 and A2<0 go to stage B
  3. Else increase the window size
  4. If window size ≤ Smax repeat stage A
  5. Else output zxy
  ```

  Stage B:
  ```
  1. Compute B1=zxy-zmin, B2=zxy-zmax
  2. If B1>0 and B2<0 output zxy
  3. Else output zmed
  ```

- Similarity measures are compared in the following way: firstly, we compute their values between the original image and the image with noise; next, between the original image and the image subjected to noise removal (we compare the two obtained values). These measures are computed according to the following formulas (*M*, *N* - the dimensions of the images, *f* - the first image, *f* with a hat - the second image, and the maximum value is computed w.r.t. the whole image):

- Mean square error:

$$MSE = \frac{1}{MN}\sum_{x=1}^{M}\sum_{y=1}^{N}\left[f(x,y)-\hat{f}(x,y)\right]^2$$

- Peak mean square error:

$$PMSE = \frac{1}{MN}\sum_{x=1}^{M}\sum_{y=1}^{N}\left[f(x,y)-\hat{f}(x,y)\right]^2 / [\max\{f(x,y)\}]^2$$

- Signal to noise ratio [dB]:

$$SNR = 10\log_{10}\left(\sum_{x=1}^{M}\sum_{y=1}^{N}[f(x,y)]^2 / \sum_{x=1}^{M}\sum_{y=1}^{N}\left[f(x,y)-\hat{f}(x,y)\right]^2\right)$$

- Peak signal to noise ratio [dB]:

$$PSNR = 10\log_{10}\frac{\sum_{x=1}^{M}\sum_{y=1}^{N}[\max\{f(x,y)\}]^2}{\sum_{x=1}^{M}\sum_{y=1}^{N}\left[f(x,y)-\hat{f}(x,y)\right]^2}$$

- Maximum difference:

$$MD = \max\left\{\left|f(x,y)-\hat{f}(x,y)\right|\right\}$$

- Additional remarks:
  - Every method should be applicable to both grey-scaled and colour images
  - It is worth paying attention to the effectiveness of the implemented methods w.r.t. the time of the whole operation
  - All original concepts for enhancement of the aforementioned methods and their comparison with classical solutions (based on literature or on one's own ideas) are welcome.