Write a program for image processing on GPU by using CUDA or OpenCL frameworks. You can use either C++ language or Python, depending on your preference.

Estimate the program performance. For that purpose, use the average program execution time for at least 10 launches. Compare the performance for various image sizes: 1024x768, 1280x960, 2048x1536. Compare the performance of the program on both CPU and GPU (CUDA/OpenCL). Compare the obtained results and make conclusions.

Use your journal number to select the image processing algorithm, according to your journal number in reverse order. Students with journal number 1 get the variant 5, journal number 2 – variant 4, 3 -> 3, 4 -> 2, 5 -> 1. Students with journal number 6 get the variant 5, number 7 – variant 4, etc.

I will also list some of the tutorial examples here. However, I do not guarantee that they will definitely suit your every need. CUDA Tutorial (C++), OpenCL Tutorial (C++), How to Cuda (Python), GPU Image Processing Using OpenCL (Python), PyOpenCL.

Task 1. Processing images with a Gaussian filter

Get intensity values $I_v = (Red_v + Green_v + Blue_v)/3$,
where I_v – v–th pixel intensity, Red_v – red component value of v, $Green_v$– green component of v, $Blue_v$– blue component of v.
Filter the matrix I with a Gaussian filter.

Gaussian Filter

1/16	2/16	1/16
2/16	4/16	2/16
1/16	2/16	1/16

Save the result to a file.

Task 2. Scale the image twice

Get intensity values $I_v = (Red_v + Green_v + Blue_v)/3$,
where I_v – v–th pixel intensity, Red_v – red component value of v, $Green_v$– green component of v, $Blue_v$– blue component of v.

The resulting image contains the replacement of each pixel in the initial grey image with a 2 x 2 pixel fragment with the same intensity value.

Save the scaling result to a file.

Example
Original matrix M1:

5	8

7	12

Output matrix

5	5	8	8
5	5	8	8
7	7	12	12
7	7	12	12

Task 3. Correction of color channels

Get the modified blue and yellow channels in the image according to the following expressions for the blue and yellow color components, respectively:

$$B_v = Blue_v - (Green_v + Blue_v)/2,$$
$$Y_v = Red_v + Green_v - 2 \cdot (|Red_v - Green_v| + Blue_v)$$

where I_v – v–th pixel intensity, Red_v – red component value of v, $Green_v$– green component of v, $Blue_v$– blue component of v.

Save two grayscale images corresponding to the modified blue and yellow channels to a file.

Task 4 Subtract grey-scale images

For two images of equal size, get the values of the intensity matrices I^A and I^B.
$I_v = (Red_v + Green_v + Blue_v)/3$,
where I_v – v–th pixel intensity, Red_v – red component value of v, $Green_v$– green component of v, $Blue_v$– blue component of v.
Perform pixel-by-pixel (element-by-element) matrix subtraction I^A and I^B :

$$I_v^C = I_v^A - I_v^B$$

Find the subtraction intensity matrix of grayscale images using the formula:
$I_v^{Result} = (I_v^C + 255)/2$.
Save the result matrix I^{Result} to a file.

Task 5. Image quantization

Get intensity values $I_v = (Red_v + Green_v + Blue_v)/3$,
where I_v – v–th pixel intensity, Red_v – red component value of v, $Green_v$– green component of v, $Blue_v$– blue component of v.

Set the number of quantization levels K - any integer in the range from 4 to 10 and split the intensity scale into quantization levels. For quants, match the following colors:

1st quantum RGB=(0,0,0)
2nd quantum RGB=(127,0,0)
3rd quantum RGB=(255,0,0)
4th quantum RGB=(0,127,0)
5th quantum RGB=(0,255,0)
6th quantum RGB=(0,0,127)
7th quantum RGB=(0,0,255)

8th quantum RGB=(127,0,127)
9th quantum RGB=(127,127,0)
10th quantum RGB=(0,127,127)

For each pixel, replace the color channel values with the color values of the quantum to which the intensity of this pixel belongs.
Save the resulting image to a file.

Quantization example. Quantization levels = 10. Halftone values range from 0 to 255. The range of halftones for one quant: 255/10 = 25.5. First quant (0) is in the range from 0 to 25.5 , second one(1) – from 25.5 to 51, third one (2) – from 51 to 76.5 etc.

6	15	4	55
13	23	21	123
28	29	89	122
37	47	101	106

Intensity values for matrix M1

0	0	0	2
0	0	0	4
1	1	3	4
1	1	3	4

Quantization example for matrix M1