Правительство Российской Федерации

Федеральное государственное автономное образовательное учреждение

высшего образования

«Национальный исследовательский университет

«Высшая школа экономики»»

Московский институт электроники и математики

Департамент компьютерной инженерии

**Математический компьютерный практикум.**

**Отчёт по контрольной работе.**

**Выполнил:**

Студент группы БИВ175:

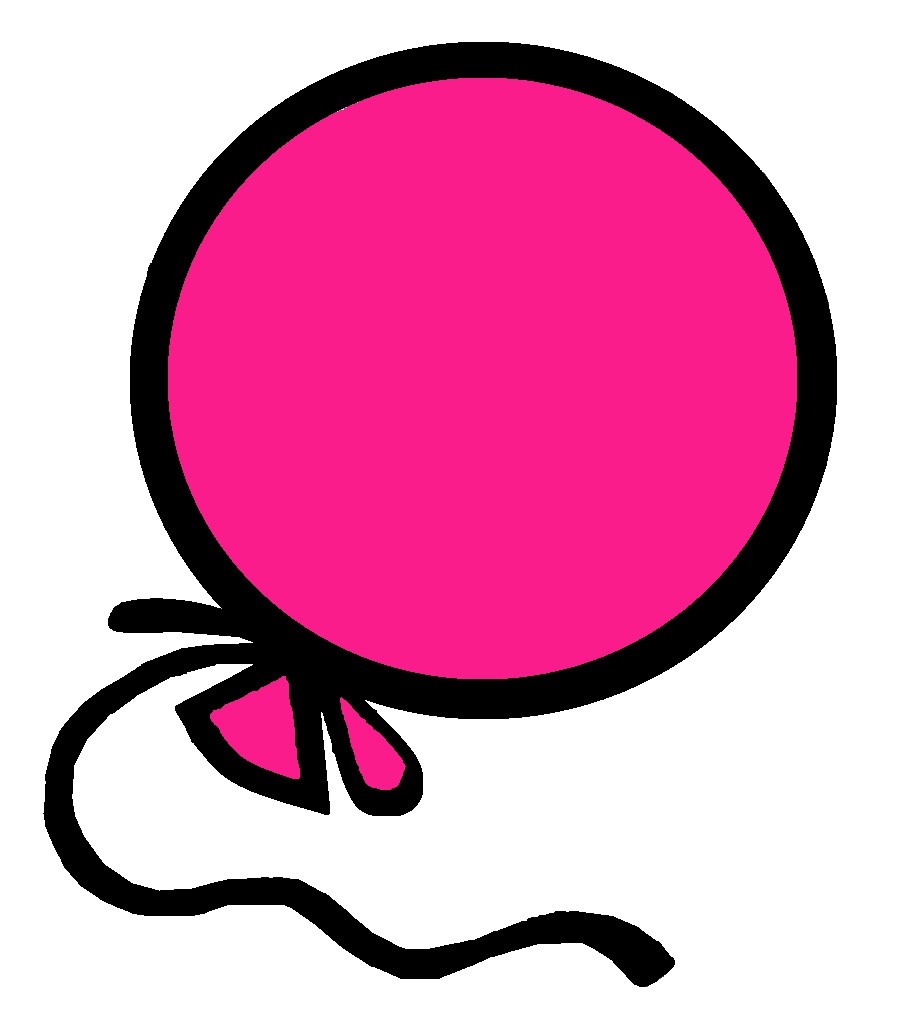
Шляк Павел Вячеславович

**Проверила:**

Байбикова Татьяна Николаевна

Исходные данные

Воздушный шарик. Нарисован знакомой.



Листинги

# Программа 1

﻿# Программа, создающая одно изображение из разных фрагментов изображений.

#Показать умение: применять аффинные преобразования к изображению,

#выполнять зеркальные отображения изображения по горизонтали и вертикали,

#масштабировать изображение и добавлять его в составное изображение.

#Результат должен быть сохранен в файл.

function main()

SRC = read\_file();

SRC(SRC==0) = 1;

B = [imrotate(I(SRC),45,'bilinear','crop') imrotate(I(SRC),27,'bilinear','crop') imrotate(I(SRC),9,'bilinear','crop') imrotate(I(SRC),-9,'bilinear','crop') imrotate(I(SRC),-27,'bilinear','crop') imrotate(I(SRC),-45,'bilinear','crop');...

imrotate(I(SRC),63,'bilinear','crop') SRC SRC SRC SRC imrotate(I(SRC),-63,'bilinear','crop');

imrotate(I(SRC),81,'bilinear','crop') SRC SRC SRC SRC imrotate(I(SRC),-81,'bilinear','crop');

imrotate(I(SRC),99,'bilinear','crop') SRC SRC SRC SRC imrotate(I(SRC),-99,'bilinear','crop');

imrotate(I(SRC),117,'bilinear','crop') SRC SRC SRC SRC imrotate(I(SRC),-117,'bilinear','crop');

imrotate(I(SRC),135,'bilinear','crop') imrotate(I(SRC),153,'bilinear','crop') imrotate(I(SRC),171,'bilinear','crop') imrotate(I(SRC),189,'bilinear','crop') imrotate(I(SRC),207,'bilinear','crop') imrotate(I(SRC),225,'bilinear','crop')];

B(B==0) = 255;

imshow(B);

imwrite(B, "file.jpg");

endfunction

function IMG = read\_file()

fn = input('File name: ', 's');

f = imfinfo(fn);

im\_type=f.ColorType

im\_indexed='indexed';

if strcmp(im\_type, im\_indexed)

[im\_1, map] = imread(fn);

IMG = ind2rgb(im\_1, map);

else

IMG = imread(fn);

endif

endfunction

function IMG = I(src)

color = randi([1 3],1,3);

IMG(:,:,[1 2 3]) = src(:,:,color);

Endfunction

# Программа 2

﻿function task2()

#Load image

[IMG, path] = read\_file();

#Detect type

detect\_type(path);

#Hist

histogramma(IMG);

#Noice

ns = noice(IMG);

#Filter

filt(ns);

#Save image

endfunction

function type = detect\_type(path)

f = imfinfo(path);

type=f.ColorType;

endfunction

function [IMG, fn] = read\_file()

fn = input('File name: ', 's');

f = imfinfo(fn);

im\_type=f.ColorType

im\_indexed='indexed';

if strcmp(im\_type, im\_indexed)

[im\_1, map] = imread(fn);

IMG = ind2rgb(im\_1, map);

else

IMG = imread(fn);

endif

endfunction

function image\_save(IMG)

fn = input('File name: ', 's');

imwrite(IMG, fn)

endfunction

function histogramma(IMG)

imhist(IMG);

title ('Histogramma');

endfunction

function noicy = noice(IMG)

noicy=rgb2gray(IMG);

noicy=imnoise(noicy,'salt & pepper');

figure;

imshow(noicy);

title('Noice');

endfunction

function D = filt (Is)

D=medfilt2(Is);

figure, imshow(D);

title('Filter')

h=fspecial('unsharp');

D=filter2(h,D);

figure, imshow(D);

title('unsharp');

endfunction

function [m,c,z] = compare (D, S)

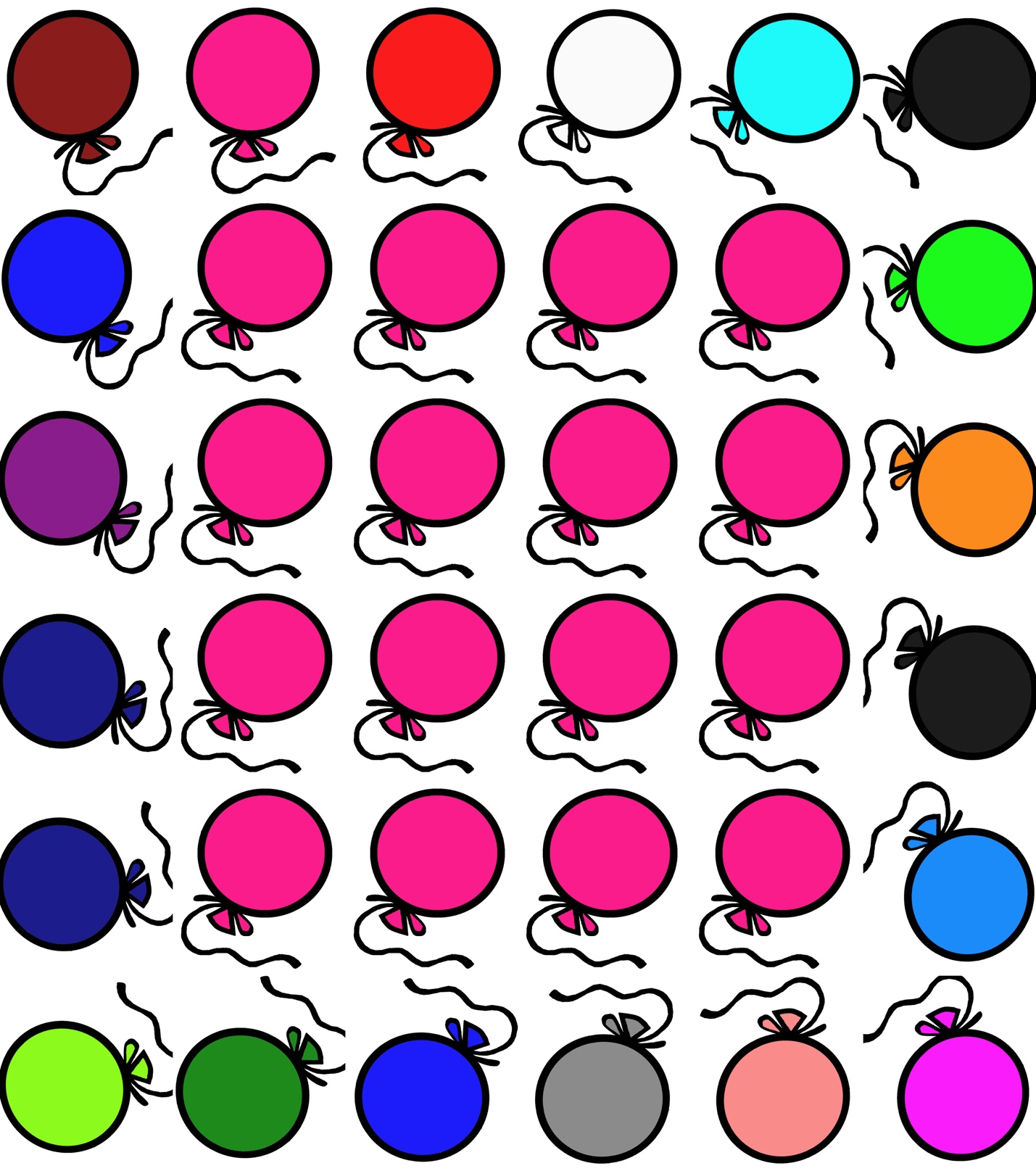
m1 = std2(S); m2 = std2(D); m=m1-m2;

c1 = mean2(S); c2 = mean2(D); c = c1-c2;

z = corr2(S,D);

endfunction

# Полученные результаты



Краткое описание используемых команд

Imrotate – поворачивает изображение

Imfinfo – получает информацию о файле

read\_file – читает файл

image\_save – сохраняет его

detect\_type – определяет тип

noice – создаёт шум

filt – филтрует шум

compare – сравнивает изображения