

Laboratorium **Programowanie w języku Python 2**
Wydział Elektrotechniki Automatyki I Informatyki
Politechnika Świętokrzyska

Studia: Stacjonarne I stopnia	Kierunek: Informatyka
Data wykonania: 27.05.2021	Grupa: 3ID16B
Imię i nazwisko: Arkadiusz Więclaw	Temat ćwiczenia: Przetwarzania obrazów

Zad 1:

```
from PIL import Image
from PIL import ImageFilter
from PIL import ImageEnhance
import os, sys

def ex_1():
    """
    Program wyswietla informacje o pliku
    """
    file1 = input("Podaj nazwe pliku: ")
    im = Image.open(file1)
    print(im.format, im.size, im.mode)
    im.show()

def exm2():
    """
    Program konwertuje obrazek na format .jpg.
    """
    file1 = input("Podaj nazwe pliku ktorego chcesz skonwertowac do
jpg: ")
    f, e = os.path.splitext(file1)
    outfile = f + ".jpg"
    if file1 != outfile:
        try:
            with Image.open(file1) as im:
                im.save(outfile)
        except OSError:
            print("cannot convert", file1)

def exm3():
    """
    Program tworzy plik JPEG thumbnails
    """
    size = (128, 128)
    infile = input("Podaj nazwe pliku ktorego chcesz skonwertowac do
jpg: ")
    outfile = os.path.splitext(infile)[0] + ".thumbnail"
    if infile != outfile:
```

```

        try:
            with Image.open(infile) as im:
                im.thumbnail(size)
                im.save(outfile, "JPEG")
        except OSError:
            print("cannot create thumbnail for", infile)
def exm4():
    """
    Program wyswietla informacje o plikach
    """
    for infile in sys.argv[1:]:
        try:
            with Image.open(infile) as im:
                print(infile, im.format, "{}x{}".format(im.size,
im.mode ))
        except OSError:
            pass
def exm5():
    """
    Przykład modyfikuje polozenie obrazka
    """
    im = Image.open("2.png")
    box = (200, 200, 400, 400)
    region = im.crop(box)
    region = region.transpose(Image.ROTATE_180)
    im.paste(region, box)
    im.show()
def exm6():
    """
    Przykład modyfikuje polozenie obrazka
    """
    im = Image.open("2.png")
    out = im.resize((130, 130))
    out = im.rotate(45) # degrees counter-clockwise
    out = im.transpose(Image.FLIP_LEFT_RIGHT)
    out = im.transpose(Image.FLIP_TOP_BOTTOM)
    out = im.transpose(Image.ROTATE_90)
    out = im.transpose(Image.ROTATE_180)
    out = im.transpose(Image.ROTATE_270)
    im.show()
if __name__ == "__main__":
    #ex_1()

```

```
#exm2()  
#exm3()  
exm4()  
exm5()  
exm6()
```

Wyniki:



Zad 2:

```
import numpy as np
import matplotlib.pyplot as plt
import skimage
from skimage import data # most functions are in subpackages
from skimage import filters
from skimage import io
import os
import scipy.misc
from skimage import exposure
```

```
def ex1():
    """
        Przykład wyświetla szachownice
    """
    check = np.zeros((10, 10))
    check[::2, 1::2] = 2
    check[1::2, ::2] = 2
    plt.matshow(check, cmap='hot')
    plt.show()
```

ex1()

```
def ex2():
    """
        Przykład pobiera plik camera i wyświetla jego właściwości.
    """
    camera = data.camera()
    print("Typ obrazu", camera.dtype)
    print("Rozmiar", camera.shape)
    filtered_camera = filters.gaussian(camera, 1)
    print(type(filtered_camera))
```

ex2()

```
def ex3():
    """
        Przykład wczytuje obrazek a następnie zapisuje go pod inną nazwą
    """
```

```
filename = os.path.join(skimage.data_dir, 'camera.png')
camera = io.imread(filename)
logo = io.imread('logo.png')
io.imsave('my_logo.png', logo)
```

```
ex3()
```

```
def ex4():
```

```
    camera = data.camera()
    camera_equalized = exposure.equalize_hist(camera)
```

```
ex4()
```

```
def ex5():
```

```
    """
```

```
    Przykład wczytuje obrazek a następnie wyswietla go na wykresie
    """
```

```
    face = scipy.misc.face()
    print(face.shape)
    plt.gray()
    plt.imshow(face)
    plt.show()
```

```
ex5()
```

```
from skimage.morphology import disk
from skimage import restoration
```

```
def ex6():
```

```
    """
```

```
    Przykład wyswietla obrazek
    """
```

```
    coins = data.coins()
    mask = coins > filters.threshold_otsu(coins)
    clean_border = segmentation.clear_border(mask)
    plt.figure()
    plt.imshow(clean_border, cmap='gray')
    plt.figure()
    plt.imshow(coins, cmap='gray')
    plt.contour(clean_border, [0.5])
```

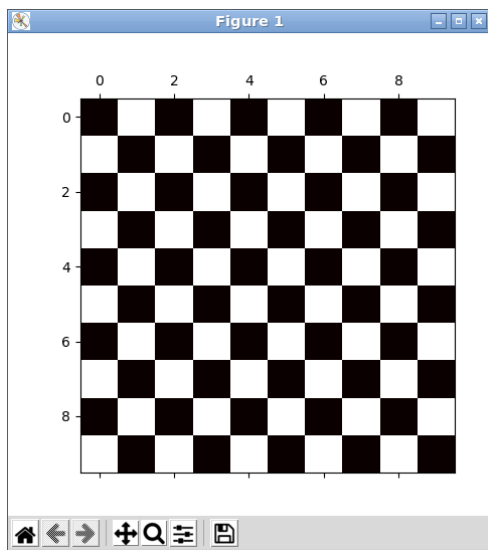
```
ex6()
```

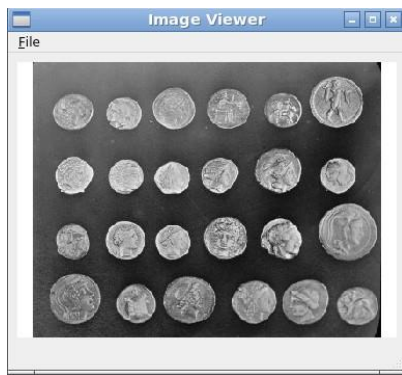
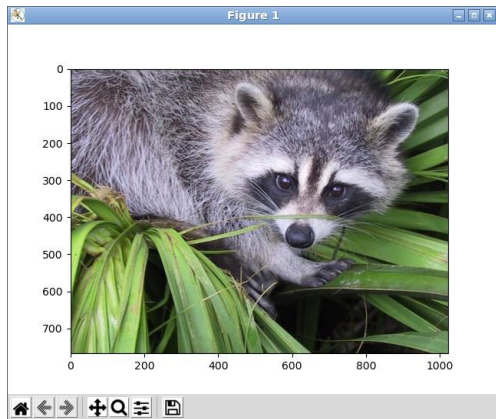
```
from skimage import viewer
from skimage.viewer.plugins import lineprofile
```

```
def ex7():
    """
    Przykład wyswietla obrazek
    """
    coins = data.coins()
    new_viewer = viewer.ImageViewer(coins)
    new_viewer.show()
    new_viewer = viewer.ImageViewer(coins)
```

```
ex7()
```

Wyniki:





Zad 3:

```
from scipy import misc
from scipy import ndimage
import numpy as np
import scipy
import imageio
import matplotlib.pyplot as plt
from glob import glob
def ex1():
    """
    Przykład zapisuje obrazek do pliku i wyswietla do na wykresie
    """
    f = misc.face()
    imageio.imsave('przyklad1.png', f) # uses the Image module (PIL)
    plt.imshow(f)
    plt.show()
def ex2():
    """
    Przykład, który wyswietla twarz szopa w matplotlib w
    rtrzechroznym odcieniach
    """
```



```

f = scipy.misc.face(gray=True)
plt.figure(figsize=(10, 3.6))
plt.subplot(131)
plt.imshow(f, cmap=plt.cm.hot)
plt.subplot(132)
plt.imshow(f, cmap=plt.cm.gray, vmin=30, vmax=200)
plt.axis('off')
plt.subplot(133)
plt.imshow(f, cmap=plt.cm.hot)
plt.contour(f, [50, 200])
plt.axis('off')
plt.subplots_adjust(wspace=0, hspace=0., top=0.99, bottom=0.01,
left=0.05,
right=0.99)
plt.show()

def ex3():
    """
    Przyklad tworzy plik raw i go otwiera
    """
    face = misc.face()
    imageio.imwrite('face.png', face) # First we need to create the PNG
file
    face = imageio.imread('face.png')
    face.tofile('face2.raw') # Create raw file
    face_from_raw = np.fromfile('face2.raw', dtype=np.uint8)
    print(face_from_raw.shape)
    face_from_raw.shape = (768, 1024, 3)
    face_memmap = np.memmap('face2.raw', dtype=np.uint8, shape=(768,
1024, 3))

def ex5():
    """
    Przyklad wyswietla informacje o pliku i tworzy tablice z pliku
obrazu
    """
    face = misc.face()
    imageio.imwrite('face.png', face) # First we need to create the PNG
file
    face = imageio.imread('face.png')
    print(type(face))
    print("informacje ", face.shape, face.dtype)

```

```

def ex6():
    """
    Wyświetla statystyczne informacje o obrazku
    """
    face = misc.face(gray=True)
    print(face.mean())
    print("minimalny rozmiar ", face.min())
    print("maksymalny rozmiar ", face.max())

def ex7():
    """
    Ten przykład pokazuje, jak manipulować obrazem przy użyciu
    typowych sztuczek numpy tablic.
    """
    face = scipy.misc.face(gray=True)
    face[10:13, 20:23]
    face[100:120] = 255
    lx, ly = face.shape
    X, Y = np.ogrid[0:lx, 0:ly]
    mask = (X - lx/2)**2 + (Y - ly/2)**2 > lx*ly/4
    face[mask] = 0
    face[range(400), range(400)] = 255
    plt.figure(figsize=(3, 3))
    plt.axes([0, 0, 1, 1])
    plt.imshow(face, cmap=plt.cm.hot)
    plt.axis('off')
    plt.show()

def ex8():
    """
    Przykład ilustruje interpolację obrazu szopa
    """
    f = scipy.misc.face(gray=True)
    plt.figure(figsize=(8, 4))
    plt.subplot(1, 2, 1)
    plt.imshow(f[320:340, 510:530], cmap=plt.cm.hot)
    plt.axis('off')
    plt.subplot(1, 2, 2)
    plt.imshow(f[320:340, 510:530], cmap=plt.cm.hot,
interpolation='nearest')
    plt.axis('off')
    plt.subplots_adjust(wspace=0.02, hspace=0.02, top=1, bottom=0,

```

```

left=0, right=1)
    plt.show()

def ex9():
    """
        Ten przykład pokazuje kilka prostych przekształceń geometrycznych
        na twarzy Szopa.
    """
    face = scipy.misc.face(gray=True)
    blurred_face = ndimage.gaussian_filter(face, sigma=3)
    very_blurred = ndimage.gaussian_filter(face, sigma=5)
    local_mean = ndimage.uniform_filter(face, size=11)
    plt.figure(figsize=(9, 3))
    plt.subplot(131)
    plt.imshow(blurred_face, cmap=plt.cm.hot)
    plt.axis('off')
    plt.subplot(132)
    plt.imshow(very_blurred, cmap=plt.cm.gray)
    plt.axis('off')
    plt.subplot(133)
    plt.imshow(local_mean, cmap=plt.cm.hot)
    plt.axis('off')
    plt.subplots_adjust(wspace=0, hspace=0., top=0.89, bottom=0.11,
        left=0.11, right=0.89)
    plt.show()

def ex10():
    """
        Ten przykład demonstruje odszumianie obrazu twarzy Szopa.
    """
    f = scipy.misc.face(gray=True)
    f = f[230:290, 220:320]
    noisy = f + 0.4*f.std()*np.random.random(f.shape)
    gauss_denoised = ndimage.gaussian_filter(noisy, 2)
    med_denoised = ndimage.median_filter(noisy, 3)
    plt.figure(figsize=(10, 3.12))
    plt.subplot(131)
    plt.imshow(gauss_denoised, cmap=plt.cm.hot, vmin=30, vmax=220)
    plt.axis('off')
    plt.title('Gaussian filter', fontsize=30)
    plt.subplot(132)
    plt.imshow(noisy, cmap=plt.cm.gray, vmin=30, vmax=220)
    plt.axis('off')

```

```
plt.title('noisy', fontsize=30)
plt.subplot(133)
plt.imshow(noisy, cmap=plt.cm.hot, vmin=30, vmax=220)
plt.axis('off')
plt.title('Median filter', fontsize=30)
plt.subplots_adjust(wspace=0.04, hspace=0.04, top=0.7, bottom=0,
left=0,
right=1)
plt.show()
```

Wyniki:

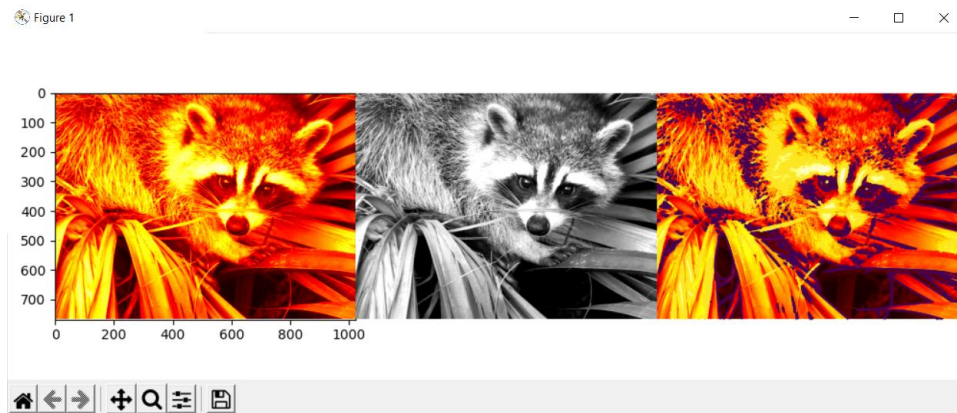
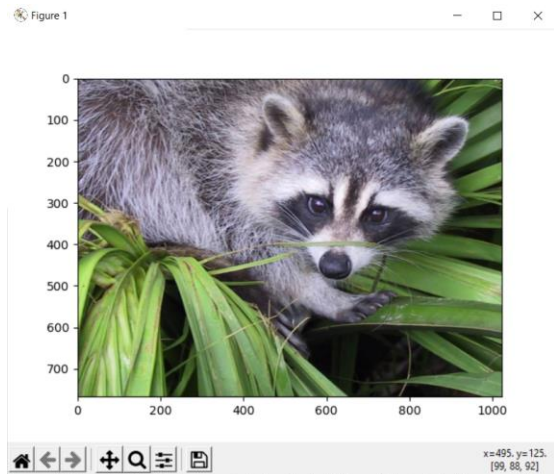


Figure 1

