

1. Wybierz planszę pt. „Przekształcenia arytmetyczne i operacje LUT”. Kliknij prawym klawiszem w obszarze „Obraz Wejściowy” i wybierz obraz „Tank”. Przekształć obraz za pomocą przekształcenia LUT polegającego na zamianie pikseli o poziomach szarości z zakresu od 41 do 67 oraz od 101 do 149 na piksele o jasności 255 oraz zamianie pikseli o innych poziomach szarości na piksele czarne o jasności 0. Ile jest **czarnych(białych)** pikseli w przekształconym obrazie ?

a)22751,b)22721,c)22472,d)22715

poprawna odpowiedź to:20879(czarne.)

44657 (białych)

```
function obraz_wy=stud_OperacjeLUT(obraz_we,tablica_lut)
%Tutaj proszę napisać funkcję realizującą
%PRZEKSZTAŁCENIA Look Up Table
%Zmienne w programie:
%obraz_we - obraz wejściowy
%tab_lut - tablica przekształceń LUT (tab_lut ma formę wektora)
%obraz_wy - obraz wyjściowy
%% Miejsce do pisania dla Studenta
a=double(obraz_we);
LUT=zeros(1,256);
LUT(42:68)=255;
LUT(102:150)=255;
obraz_wy=LUT(a+1);
%obraz_wy=tablica_lut(a+1);
%(jeśli czarne to)
b=256*256-sum(sum(sign(obraz_wy)))
obraz_wy=uint8(obraz_wy);
%(Jeśli białych to)
B=sum(sum(sign(obraz_wy)))
obraz_wy=uint8(obraz_wy);
```

%%Koniec tego miejsca - NIC WIĘCEJ NIE ZMIENIAĆ !!!

2. Wybierz planszę pt. „Filtr liniowy”. Kliknij prawym klawiszem w obszarze „Obraz Wejściowy” i wybierz obraz „Kobieta”. Przekształć obraz za pomocą filtracji górnoprzepustowej z normalizacją wg wzoru:

$A_1(m,n) = \text{round}(255 \cdot (A(m,n) - \min A) / (\max A - \min A))$

gdzie:

minA – minimalna wartość w obrazie poddanym konwolucji,

maxA – maksymalna ...

A(m,n) – piksel o współrzędnych m, n obrazu wyjściowego,

A₁(m,n) – piksel ... wejściowego.

Ile jest pikseli o poziomach szarości z zakresu od 101:103 w znormalizowanym obrazie A₁ jeżeli użyto maski filtru górnoprzepustowego:

$$Maska = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

a)903,b)904,c)901,d)902

W szablonie funkcji jako argument wejściowy używana jest zmienna maska9. Jest to wektor wierszowy (1x9), którego pierwsze 3 elementy stanowią 1 wiersz maski 3x3, kolejne 3 drugi wiersz maski 3x3 i ostatnie 3 ostatni wiersz maski 3x3.

```
function obraz_wy=stud_Maska9(obraz_we,maska9)
```

```
%Tutaj proszę napisać funkcję realizującą
```

```
%FILTR LINIOWY - MASKA 3x3
```

```
%Zmienne w programie:
```

```
%obraz_we - obraz wejściowy
```

```
%maska9 - maska 9 elementowa (macierz [1,9])
```

```
%obraz_wy - obraz wyjściowy
```

```
%% Miejsce do pisania dla Studenta
```

```
%Poszerzanie obrazu
```

```
obraz_we=double(obraz_we);
```

```
LW=size(obraz_we,1);
```

```
LK=size(obraz_we,2);
```

```
temp=zeros(LW+2,LK+2);
```

```
%zadeklarowanie powiększonej macierzy z obrazem
```

```
temp(2:LW+1,2:LK+1)=obraz_we;
```

```
%wpisanie obrazu wejściowego w środek powiększonej macierzy
```

```
temp(1,2:LK+1)=obraz_we(1,:);
```

```
%powielenie pierwszego wiersza
```

```
temp(LW+2,2:LK+1)=obraz_we(LW,:);
```

```
%powielenie ostatniego wiersza
```

```
temp(:,1)=temp(:,2);
```

```
%powielenie pierwszej kolumny
```

```
temp(:,LK+2)=temp(:,LK+1);
```

```
%powielenie ostatniej kolumny
```

```
%Od tego miejsca należy dopisać własne linie programu
```

```
B=zeros(LW,LK);
```

```
for i=2:size(temp,1)-1
```

```

for j=2:size(temp,2)-1
    B(i-1,j-1)=sum(sum(temp(i-1:i+1,j-1:j+1).*[-1 -1 -1; -1 8 -1; -1 -1 -1]));
end;
end;
B=round(255*(B-min(min(B)))/(max(max(B))-min(min(B))));

obraz_wy=uint8(B);

```

- Wybierz planszę pt. „Filtr liniowy”. Kliknij prawym klawiszem w obszarze „Obraz Wejściowy” i wybierz obraz „Łódka”. Przekształć obraz za pomocą filtra uśredniającego 3x3 z maską

$$Maska = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

jaka jest minimalna wartość jasności piksela w obrazie ?

- 218
- 11**
- 217
- 12

```
function obraz_wy=stest_gr3b_2(obraz_we)
```

```

% 2) Obraz "boat256" przefiltrowano filtrem uśredniającym, jaka jest minimalna wartość
% jasności piksela w obrazie ? Przed filtracją powiększyć obraz wejściowy przez dodanie
% linii zer z każdej strony obrazu w celu uniknięcia problemów z filtracją na brzegach
% obrazu.

```

```

a=imread(obraz_we);
a=double(a);
W=size(a,1); % liczba wierszy a
K=size(a,2); % liczba kolumn a

a1=zeros(W+2,K+2);
obraz_wy=zeros(W,K);
size(a1)
a1(2:W+1,2:K+1)=a;

for i=2:size(a1,1)-1
    for j=2:size(a1,2)-1
        obraz_wy(i-1,j-1)=round(sum(sum(a1(i-1:i+1,j-1:j+1)))/9);
    end;
end;

min(min(obraz_wy))

```

- Wybierz planszę „Dwuwymiarowa dyskretna transformata kosinusowa”. Następnie dokonaj odwrotnej dwuwymiarowej DCT obrazu 8 x 8 składającego się z białego piksela o jasności 255 w punkcie (2,2) oraz czarnych pikseli o jasności 0 w pozostałych punktach. Wartości pikseli można ustawić klikając prawym klawiszem myszy na obszar „Obraz Transformaty” i wybierając opcję ustaw „współczynniki DCT”. Jeżeli dany element zostanie zaznaczony kropką, to przyjmuje wartość 255, w przeciwnym przypadku ma wartość 0. Który punkt obrazu wyjściowego ma wartość 61,3237.

- (1,8)
- (1,1)**
- (8,1)
- (2,2)

W celu wykonania tego zadania należy ustawić opcję ‘File/Preferences/Numeric format’ na wartość short w oknie głównym MATLAB-a. Następnie w funkcji „Tworzenie współczynników macierzy DCT” należy obliczyć macierz D wg wzoru:

$$D_{ab} = \sqrt{\frac{2}{N}} C_a \cos \frac{a\pi(b+0.5)}{N}, \quad (1)$$

gdzie: $C_a = 1/\sqrt{2}$ dla $a=0$ lub $C_a=1$ w przeciwnym przypadku

a – indeks wiersza,

b – indeks kolumny.

Następnie w funkcji „Odwrotna dyskretna transformata kosinusowa” Należy napisać funkcję na obliczanie IDCT2.

```

function D=stud_MacierzWspolczynn timerowDCT

% Funkcja ta powinna przygotowywać macierz D współczynników dyskretnej transformaty
kosinusowej
% do dalszej transformaty bloku obrazu o wymiarze 8/8
%-----
% Aby to wykonać należy stworzyć plik o nazwie: stud_MacierzWspolczynn timerowDCT.m
% Pierwsza linijka pliku powinna zawierać definicję funkcji łącznie z jej nazwą, w tym
przypadku powinno wyglądać:
% function D=stud_MacierzWspolczynn timerowDCT

%*****
% ***** MIEJSCE DO PISANIA DLA STUDENTA *****

%Przykładowe rozwiązanie
N=8;
for a=0:N-1
    if a==0
        Ca=1/(2^(1/2));
    else
        Ca=1;
    end
    for b=0:N-1
        D(a+1,b+1)=(2/N)^(1/2)*Ca*cos(a*pi*(b+0.5)/N);
    end
end

% ***** KONIEC NIC WIECEJ NIE ZMIENIAC *****
%*****

function ObrazWyj = stud_TransformataIDCT(MacierzDct)

% Funkcja ta powinna obliczać odwrotną transformatę DCT z macierzy 8x8
% Potrzebną macierz współczynników DCT otrzymamy wywołując napisaną
% wcześniej funkcję na tworzenie macierzy współczynników DCT. Wywołujemy ją
% następująco: D=stud_MacierzWspolczynn timerowDCT. Funkcja ta powinna zwracać
% nam gotowe współczynn timerowiki, które wykorzystamy do przeprowadzenia
% odwrotnej transformaty DCT obrazu.
%
% Zmienne w programie:
% MacierzDct      - macierz będąca wynikiem transformaty DCT
% ObrazWyj         - obraz odtworzony po odwrotnej DCT
% D - macierz wsp. DCT o rozmiarze 8/8
%
%-----
% Aby to wykonać należy stworzyć plik o nazwie: stud_TransformataIDCT.m
% Pierwsza linijka pliku powinna zawierać definicję funkcji łącznie z jej nazwą,
% w tym przypadku powinno wyglądać:
% function ObrazWyj = stud_TransformataIDCT(MacierzDct)

%*****
% ***** MIEJSCE DO PISANIA DLA STUDENTA *****

%Przykładowe rozwiązanie
D=stud_MacierzWspolczynn timerowDCT;
ObrazWyj=uint8(D' * MacierzDct * D);

% ***** KONIEC NIC WIECEJ NIE ZMIENIAC *****
%*****

```

4. Wybierz planszę pt. „Przekształcenia arytmetyczne i operacje LUT”. Kliknij prawym klawiszem w obszarze „Obraz Wejściowy” i wybierz obraz „Pentagon”. Przekształć obraz za pomocą przekształcenia arytmetycznego polegającego na dodaniu do obrazu liczby 10. Ile pikseli o poziomach szarości w zakresie od 71 do 150 zawiera obraz wyjściowy. (!!podobne zadanie zamiast dodać 10 to jest inwersja i 41 do 149, obraz mężczyzna!)

a)43606 b)42681 c)41766 d)41676

(!!!inwersja a)35964 b)36248 c)36249 d)35627 e)żadna z powyższych odp. prawidłowa odp. to:

```

function obraz_wy=stud_ArytmetyczneDodajLiczbe(obraz_we,liczba_dodawana)

```

```

%Tutaj proszę napisać funkcję realizującą
%PRZEKSZTAŁCENIA ARYTMETYCZNE - DODANIE LICZBY DO OBRAZU

```

```
%Zmienne w programie:
%obraz_we - obraz wejściowy
%obraz_wy - obraz wyjściowy

%% Miejsce do pisania dla Studenta
obraz_we=double(obraz_we);
obraz_wy=obraz_we+10;
hist=zeros(1,256);
for i=1:256
    for j=1:256
        hist(obraz_wy(i,j)+1)=hist(obraz_wy(i,j)+1)+1;
    end;
end;
L=sum(hist(72:151))
obraz_wy=uint8(obraz_wy);
%%Koniec tego miejsca - NIC WIĘCEJ NIE ZMIENIAĆ !!!
(!!! Wersja 2 inwersja)
function obraz_wy=stud_OperacjeLUT(obraz_we,tablica_lut)

%Tutaj proszę napisać funkcję realizującą
%PRZEKSZTAŁCENIA Look Up Table
%Zmienne w programie:
%obraz_we - obraz wejściowy
%tab_lut - tablica przekształceń LUT (tab_lut ma formę wektora)
%obraz_wy - obraz wyjściowy
```

```
%% Miejsce do pisania dla Studenta

a=double(obraz_we);
LUT=[255:-1:0]; % można również wybrać gotową tablicę
obraz_wy=LUT(a+1);

hist=zeros(1,256);
for i=1:256
    for j=1:256
        hist(obraz_wy(i,j)+1)=hist(obraz_wy(i,j)+1)+1;
    end;
end;
L=sum(hist(42:150)) %liczba pikseli o jasnościach z zakresu od 41 do 149

obraz_wy=uint8(obraz_wy);

%%Koniec tego miejsca - NIC WIĘCEJ NIE ZMIENIAĆ !!!
```

5. Wybierz planszę pt. „Filtr liniowy”. Kliknij prawym klawiszem w obszarze „Obraz Wejściowy” i wybierz obraz „Łódka”. Przekształć obraz za pomocą filtru uśredniającego 3x3. jaka jest minimalna wartość jasności piksela w obrazie ?

a) 23 b) 16 c) 2 d) 12 (wersja 2 -> łódka, maksymalna -> odp 217)

W szablonie funkcji jako argument wejściowy używana jest zmienna maska9. Jest to wektor wierszowy (1x9), którego pierwsze 3 elementy stanowią 1 wiersz maski 3x3, kolejne 3 drugi wiersz maski 3x3 i ostatnie 3 ostatni wiersz maski 3x3.

```
function obraz_wy=stud_Maska9(obraz_we,maska9)
%Tutaj proszę napisać funkcję realizującą
%FILTR LINIOWY - MASKA 3x3
%Zmienne w programie:
%obraz_we - obraz wejściowy
%maska9 - maska 9 elementowa (macierz [1,9])
%obraz_wy - obraz wyjściowy

%% Miejsce do pisania dla Studenta
%Poszerzanie obrazu
obraz_we=double(obraz_we);
LW=size(obraz_we,1);
LK=size(obraz_we,2);
temp=zeros(LW+2,LK+2); %zadeklarowanie powiększonej macierzy z obrazem
temp(2:LW+1,2:LK+1)=obraz_we; %wpisanie obrzau wejściowego w środek powiększonej macierzy
temp(1,2:LK+1)=obraz_we(1,:); %powielenie pierwszego wiersz
temp(LW+2,2:LK+1)=obraz_we(LW,:); %powielenie ostatniego wiersza
temp(:,1)=temp(:,2); %powielenie pierwszej kolumny
temp(:,LK+2)=temp(:,LK+1); %powielenie ostatniej kolumny
%Od tego miejsca należy dopisać własne linie programu
```

```
%a1(2:W+1,2:K+1)=a;

maska9
for i=2:size(temp,1)-1
    for j=2:size(temp,2)-1
        obraz_wy(i-1,j-1)=round(sum(sum(temp(i-1:i+1,j-1:j+1)))/9);
    end;
end;

obraz_wy=uint8(obraz_wy)
size(obraz_wy)
min(min(obraz_wy)) %wersja 2 !!! zmienic na max(max(obraz_wy))
%%Koniec tego miejsca - NIC WIĘCEJ NIE ZMIENIAĆ !!!
```

6. Wybierz planszę pt. „Dwuwymiarowa dyskretna transformata Fouriera – DFT2”. Kliknij prawym klawiszem w obszarze „Obraz Wejściowy” i wybierz obraz „**Barbara**”. Obraz przekształć za pomocą dwuwymiarowej DFT. Jaką wartość ma logarytm dziesiętny modułu elementu (127,127) F-obrazu **po** transformacji optycznej?
a)5.9084 b)2.9540 c)**5.0362** d)3.2623

```
function [macierz_f2,obraz_f2,faza_f2]=stud_TransformataFFT2(obraz_we)

%Tutaj proszę napisać funkcję realizującą
%TRANSFORMACJĘ FFT2 OBRAZU
%Zmienne w programie:
%obraz_we - obraz wejściowy
%macierz_f2 - wynik transformacji FFT2
%obraz_f2 - obraz po transformacji optycznej FFT2
%faza_f2 - faza FFT2
%W związku ze "macierz_f2" proszę zajrzeć do pomocy pod hasło: 'fft2'

%% Miejsce do pisania dla Studenta
a=double(obraz_we);
F=fft(fft(a.'').');
macierz_f2=fftshift(F);
obraz_wy=log10(abs(macierz_f2(127,127))+1)
obraz_f2=0;
faza_f2=0;

%%Koniec tego miejsca - NIC WIĘCEJ NIE ZMIENIAĆ !!!
```

7. Wybierz planszę pt. „Przekształcenia arytmetyczne i operacje LUT”. Kliknij prawym klawiszem w obszarze „Obraz Wejściowy” i wybierz obraz „Samolot”. Przekształć obraz za pomocą przekształcenia LUT typu „odwrócone V”(tzn liteta”V”). Ile jest pikseli w obrazie o poziomach szarości z zakresu od 30 do 100 ? (Lotnisko 40-151)

Odp.: 22536 (wersja 2-> **41395**)

```
function obraz_wy=stud_OperacjeLUT(obraz_we,tablica_lut)

%Tutaj proszę napisać funkcję realizującą
%PRZEKSZTAŁCENIA Look Up Table
%Zmienne w programie:
%obraz_we - obraz wejściowy
%tab_lut - tablica przekształceń LUT (tab_lut ma formę wektora)
%obraz_wy - obraz wyjściowy

%% Miejsce do pisania dla Studenta

a=double(obraz_we);

obraz_wy=tablica_lut(a+1);

hist=zeros(1,256);
for i=1:256
    for j=1:256
        hist(obraz_wy(i,j)+1)=hist(obraz_wy(i,j)+1)+1;
    end;
end;
L=sum(hist(31:101)) %liczba pikseli o jasnościach z zakresu od 30 do 100(werjsa 2!!! 41:152)

obraz_wy=uint8(obraz_wy);
```

%%Koniec tego miejsca - NIC WIĘCEJ NIE ZMIENIAĆ !!!

1. Wybierz planszę „Dwuwymiarowa dyskretna transformata kosinusowa”. Następnie dokonaj odwrotnej dwuwymiarowej DCT obrazu 8 x 8 składającego się z białego piksela o jasności 255 w punkcie (2,2) oraz czarnych pikseli o jasności 0 w pozostałych punktach. Wartości pikseli można ustawić klikając prawym klawiszem myszy na obszar „Obraz Transformaty” i wybierając opcję ustaw „współczynniki DCT”. Jeżeli dany element zostanie zaznaczony kropką, to przyjmuje wartość 255, w przeciwnym przypadku ma wartość 0. Jaką wartość posiada punkt (2,2) obrazu wyjściowego.
- e) 61,3237
 - f) 255**
 - g) 12,4567
 - h) 23,3456
 - i) żadna z powyższych odpowiedzi, prawidłowa odpowiedź to

W celu wykonania tego zadania należy ustawić opcję ‘File/Preferences/Numeric format’ na wartość short w oknie głównym MATLAB-a. Następnie w funkcji „Tworzenie współczynników macierzy DCT” należy obliczyć macierz D wg wzoru:

$$D_{ab} = \sqrt{\frac{2}{N}} C_a \cos \frac{a\pi(b+0.5)}{N}, \quad (1)$$
$$a = 0,1,\dots,7; \quad b = 0,1,\dots,7;$$

gdzie: $C_a = 1/\sqrt{2}$ dla $a=0$ lub $C_a=1$ w przeciwnym przypadku

a – indeks wiersza,

b – indeks kolumny.

Następnie w funkcji „Odwrotna dyskretna transformata kosinusowa” Należy napisać funkcję na obliczanie IDCT2.

```
function D=stud_MacierzWspolczynnikiDCT
```

```
% Funkcja ta powinna przygotowywać macierz D współczynników dyskretny transformaty kosinusowej
```

```
% do dalszej transformaty bloku obrazu o wymiarze 8/8
```

```
%-----
```

```
% Aby to wykonać należy stworzyć plik o nazwie: stud_MacierzWspolczynnikiDCT.m
```

```
% Pierwsza linijka pliku powinna zawierać definicję funkcji łącznie z jej nazwą, w tym przypadku powinno wyglądać:
```

```
% function D=stud_MacierzWspolczynnikiDCT
```

```
%*****
```

```
% ***** MIEJSCE DO PISANIA DLA STUDENTA *****
```

```
%Przykładowe rozwiązanie
```

```
N=8;
```

```
for a=0:N-1
```

```
    if a==0
```

```
        Ca=1/(2^(1/2));
```

```
    else
```

```
        Ca=1;
```

```
    end
```

```
    for b=0:N-1
```

```
        D(a+1,b+1)=(2/N)^(1/2)*Ca*cos(a*pi*(b+0.5)/N);
```

```
    end
```

```
end
```

```
% ***** KONIEC NIC WIECEJ NIE ZMIENIAC *****
```

```
%*****
```

```
function ObrazWyj = stud_TransformataIDCT(MacierzDct)
```

```
% Funkcja ta powinna obliczać odwrotną transformatę DCT z macierzy 8x8
```

```
% Potrzebną macierz współczynników DCT otrzymamy wywołując napisaną
```

```
% wcześniej funkcję na tworzenie macierzy współczynników DCT. Wywołujemy ją
```

```
% następująco: D=stud_MacierzWspolczynnikiDCT. Funkcja ta powinna zwracać
```

```
% nam gotowe współczynniki, które wykorzystamy do przeprowadzenia
```

```
% odwrotnej transformaty DCT obrazu.
```

```
%
```

```
% Zmienne w programie:
```

```
% MacierzDct      - macierz będąca wynikiem transformaty DCT
% ObrazWyj        - obraz odtworzony po odwrotnej DCT
% D - macierz wsp. DCT o rozmiarze 8/8
%
%-----
% Aby to wykonać należy stworzyć plik o nazwie: stud_TransformataIDCT.m
% Pierwsza linijka pliku powinna zawierać definicję funkcji łącznie z jej nazwą,
% w tym przypadku powinno wyglądać:
% function ObrazWyj = stud_TransformataIDCT(MacierzDct)
```

```
%*****
% ***** MIEJSCE DO PISANIA DLA STUDENTA *****
```

%Przykładowe rozwiązanie

```
D=stud_MacierzWspolczynnikowDCT;
ObrazWyj=uint8(D' * MacierzDct * D);
```

```
% ***** KONIEC NIC WIECEJ NIE ZMIENIAC *****
```

```
%*****
```

1. Wybierz planszę „Dwuwymiarowa dyskretna transformata Fouriera – DFT2”. Kliknij prawym klawiszem w obszarze „Obraz Wejściowy” i wybierz obraz „Tank”. Napisz funkcję na dokonywanie dwuwymiarowej FFT. Ile wynosi faza punktu o współrzędnych (9,71) F-obrazu **przed** dokonaniem transformacji optycznej.(wersja2!!! Mężczyzna (27,45) **po** dokonaniu)(wersja3->log dziesiętny)
Odp 0.8268 rad

Rozwiązanie:

```
function obraz_wy=stest_gr1a_3(obraz_we)    (tu może się coś zmienić)
```

```
%      3) Dokonać dwuwymiarowej FFT na obrazie. Ile wynosi faza punktu
%      o współrzędnych (127,127) F-obrazu przed dokonaniem transformacji optycznej.
%a=imread(obraz_we);    (ta linijka tylko jeśli będzie wersja 2!!! „po” w 1 wersji jej nie ma)
a=double(a);
F=fft(fft(a.').');
%F=fftshift(F);    (ta linijka tylko w 2 wersji „po”)
obraz_wy=angle(F(9,71))    (dane zmienić na podane w treści (27,45))
```

```
%(jeżeli wersja 3 to zmienić ostatnią linijkę na)
obraz_wy=log10(abs(F(127,127)))
```

1. Wybierz planszę pt. „Dwuwymiarowa dyskretna transformata Fouriera – DFT2”. Kliknij prawym klawiszem w obszarze „Obraz Wejściowy” i wybierz obraz „Barbara”. Obraz przekształć za pomocą dwuwymiarowej DFT. Jaką wartość ma logarytm dziesiętny modułu elementu (127,127) F-obrazu po transformacji optycznej?
 - a. 5.9084
 - b. 2.9540
 - c. 5.0362**
 - d. 3.2623

```
function [macierz_f2,obraz_f2,faza_f2]=stud_TransformataFFT2(obraz_we)
```

```
%Tutaj proszę napisać funkcję realizującą
%TRANSFORMACJĘ FFT2 OBRAZU
%Zmienne w programie:
%obraz_we - obraz wejściowy
%macierz_f2 - wynik transformacji FFT2
%obraz_f2 - obraz po transformacji optycznej FFT2
%faza_f2 - faza FFT2
%W związku ze "macierz_f2" proszę zajrzeć do pomocy pod hasło: 'fft2'
```

```
%% Miejsce do pisania dla Studenta
a=double(obraz_we);
F=fft(fft(a.').');
macierz_f2=fftshift(F);
obraz_wy=log10(abs(macierz_f2(127,127)))
obraz_f2=0;
faza_f2=0;
```

```
%Koniec tego miejsca - NIC WIĘCEJ NIE ZMIENIAĆ !!!
```

1. Wybierz planszę pt. „Przekształcenia arytmetyczne i operacje LUT”. Kliknij prawym klawiszem w obszarze „Obraz Wejściowy” i wybierz obraz „Pentagon”. Przekształć obraz za pomocą operacji podnoszenia do sześcienu oraz normalizacji przez podzielenie przez (255×255) oraz zaokrąglanie funkcją round. Jaka jest suma jasności pikseli w obrazie wyjściowym ?
- 2638948
 - 4058369
 - 4200345
 - 2660789**
 - żadna z powyższych odpowiedzi, prawidłowa odpowiedź to:

```
a=double(obraz_we);  
a=round(a.^3/(255*255));  
obraz_wy=uint8(a);  
sum(sum(obraz_wy))
```

1. Dokonać przekształcenia obrazu „liście” ze standardu RGB na standard CMY. Jaka jest wartość sumy jasności pikseli składowej B dla obrazu składowej Y na ekranie komputera.

```
function obraz_wy = stud_SkladoweCMY(obraz_we,kolor)
```

```
%STUD_SKLADOWECCMY Funkcja realizuje wyświetlanie składowych
```

```
% obrazu CMY oraz dopełnienie obrazu RGB.
```

```
% Zmienna 'KOLOR' decyduje o wyświetlanej składowej.
```

```
% Dla 'KOLOR' = 1 - wyświetlenie składowej C
```

```
% = 2 - wyświetlenie składowej M
```

```
% = 3 - wyświetlenie składowej Y
```

```
% = 4 - wyświetlenie dopełnienia RGB
```

```
%
```

```
% Wywołanie funkcji
```

```
% STUD_SKLADOWECCMY(OBRAZ_WE,KOLOR)
```

```
% Tutaj proszę napisać funkcję realizującą
```

```
% wyświetlanie składowych obrazu CMY oraz dopełnienie do koloru RGB
```

```
% Zmienne w programie:
```

```
% obraz_we - obraz wejściowy
```

```
% obraz_wy - obraz wyjściowy
```

```
% kolor - decyduje jaka składowa obrazu CMY będzie wyświetlana
```

```
% wartości zmiennej kolor:
```

```
%
```

1 - dla składowej C

```
%
```

2 - dla składowej M

```
%
```

3 - dla składowej Y

```
%
```

4 - dopełnienie RGB

```
%
```

```
%% Miejsce do pisania dla Studenta
```

```
obraz_we = double(obraz_we);
```

```
w = size(obraz_we);
```

```
obraz_wy = zeros(w(1),w(2),w(3));
```

```
suma = 0;
```

```
for i = 1:w(1)
```

```
    for j = 1:w(2)
```

```
        obraz_wy(i,j,1) = 255;
```

```
        obraz_wy(i,j,2) = 255;
```

```
        obraz_wy(i,j,3) = 255- obraz_we(i,j,3);
```

```
        suma = suma + obraz_wy(i,j,3);
```

```
    end
```

```
end
```

```
suma
```

```
obraz_wy = uint8(obraz_wy);
```


