Prednaska 1

```
% Matlab: více slidy (Martin Trnecka) <https://www.dropbox.com/s/idbyumx5xgi339a/matlab.pdf?dl=
% https://www.dropbox.com/s/idbyumx5xgi339a/matlab.pdf?dl=0>

% Nápověda k příkazům
% help prikaz
help sum
```

```
 sum Sum of elements.
    S = sum(X) is the sum of the elements of the vector X. If X is a matrix,
    S is a row vector with the sum over each column. For N-D arrays,
    sum(X) operates along the first non-singleton dimension.

    S = sum(X,'all') sums all elements of X.

    S = sum(X,DIM) sums along the dimension DIM.

    S = sum(X,VECDIM) operates on the dimensions specified in the vector
    VECDIM. For example, sum(X,[1 2]) operates on the elements contained in
    the first and second dimensions of X.

    S = sum(...,OUTTYPE) specifies the type in which the
    sum is performed, and the type of S. Available options are:

    'double'    -  S has class double for any input X
    'native'    -  S has the same class as X
    'default'   -  If X is floating point, that is double or single,
                   S has the same class as X. If X is not floating point,
                   S has class double.

    S = sum(...,NANFLAG) specifies how NaN (Not-A-Number) values are
    treated. The default is 'includenan':

    'includenan' - the sum of a vector containing NaN values is also NaN.
    'omitnan'    - the sum of a vector containing NaN values
                   is the sum of all its non-NaN elements. If all
                   elements are NaN, the result is 0.

    Examples:
        X = [0 1 2; 3 4 5]
        sum(X, 1)
        sum(X, 2)

        X = int8(1:20)
        sum(X)              % returns double(210), accumulates in double
        sum(X,'native')     % returns int8(127), because it accumulates in
                            % int8 but overflows and saturates.

    See also prod, cumsum, diff, accumarray, isfloat.

    Documentation for sum
    Other functions named sum
```

Práce s obrázky

```
% nacteni obrazku
% imread(cesta);
I=imread('pastelky_gray.png');
```

```
% zobrazeni obrazku
% imshow(obrazek);
imshow(I);
```

roztazeni intenzit (vsechny hodnoty mensi nez low budou nastaveny na 0, vsechny vetsi nez height na 255, a hodnoty mezi low a height budou roztazeny pravidelne mezi 0 a 255 imshow(obrazek, [low,high]); [] na 0 je nastavena nejnizsi hodnota a na 255 nejvyssi

```
low = 48
```

```
low = 48
```

```
high = 212
```

```
high = 212
```

```
imshow(I,[low,high]);
```

hodnota pixelu na souradnicich 1, 1 (funkce display vypise vystup) I(x,y) pozor indexuje se od 1

```
display(I(1,1));
```

```
  uint8

   248
```

```

% velikost obrazku, h - vyska obrazku (pocet radku), w - sirka obrazku
% (pocet sloupcu), o - dimenze urcujici kolik hodnot je potreba k
% reprezentaci informace o obraze, barevny = 3, sedotonovy = 1)
% size(promenna)

[h,w,o] = size(I);
display(h);
```

```
h = 426
```

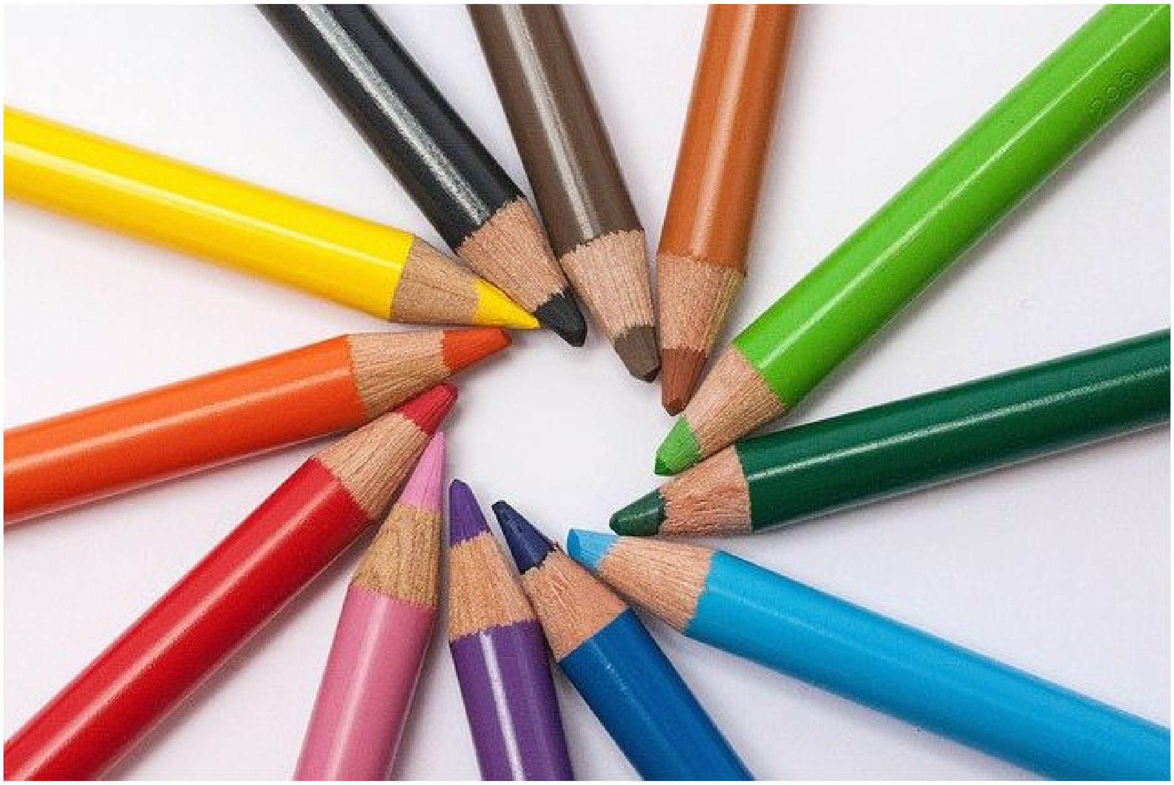
```
display(w);
```

```
w = 640
```

```
display(o);
```

```
o = 1
```

```
I_rgb = imread("pastelky.png");
figure, imshow(I_rgb);
```



```
display(I_rgb(1,1,:));
```

```
  1×1×3 uint8 array

(:,:,1) =

   248


(:,:,2) =

   248


(:,:,3) =

   250
```

```
[h,w,o] = size(I);
display(h);
```

```
h = 426
```

```
display(w);
```

```
w = 640
```

```
display(o);
```

```
o = 1
```

Vytvoreni vyrezu obrazku I(xmin:xmax, ymin:ymax)

```
xmin = 100
```

```
xmin = 100
```

```
xmax = 250
```

```
xmax = 250
```

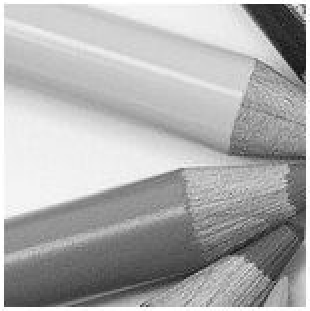
```
ymin = 100
```

```
ymin = 100
```

```
ymax = 250
```

```
ymax = 250
```

```
J = I(xmin:xmax,ymin:ymax);

imshow(J);
```



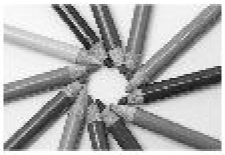
vykresleni vice obrazku do noveho okna figure, imshow()

```
imshow(I);
```

```
figure, imshow(J);
```

vykresleni vice obrazku do jednoho subplot(pocetx, pocety, pozice), imshow()

```
subplot(1,2,1), imshow(I);
subplot(1,2,2), imshow(J);
```

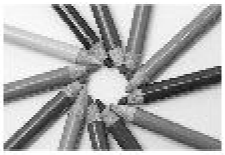


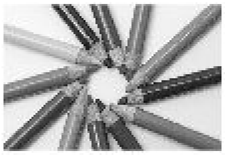
ulozeni obrazku imwrite(obrazek,cesta)

```
imwrite(J,"pastelky_vyrez.png");
```

export obrazku exportgraphics(objekt,cesta);

```
imshow(I);
f = gcf;
exportgraphics(f,'pastelky.pdf');
```





informace o obrazku imfinfo(cesta)

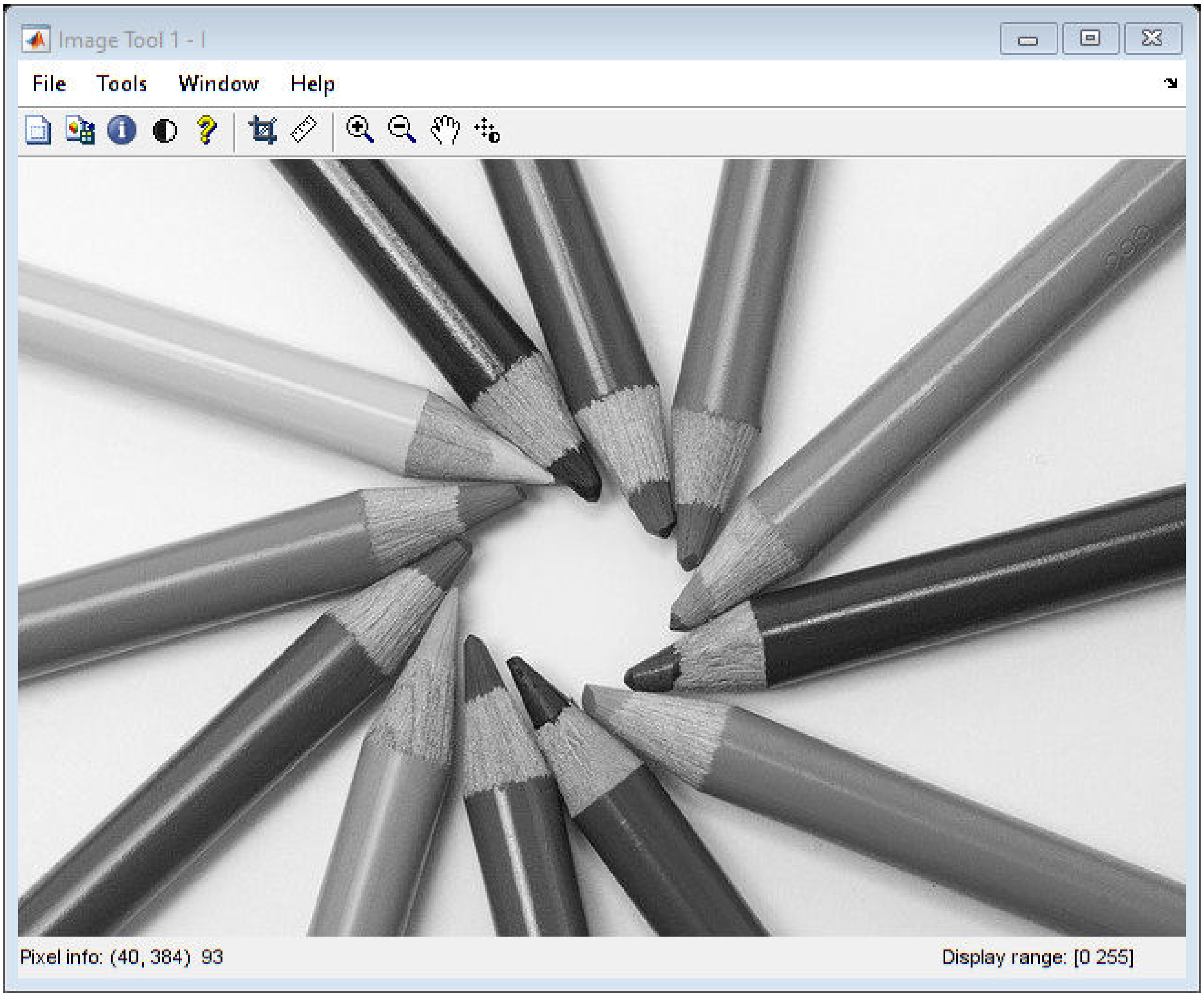
```
imfinfo('pastelky.png')
```

```
ans = struct with fields:
             Filename: 'C:\Skola\vyuka2022-2023\ZS\POGR\matlab\pr1\pastelky.png'
          FileModDate: '01-Mar-2022 19:16:55'
             FileSize: 449411
               Format: 'png'
        FormatVersion: []
                Width: 640
```

```
                 Height: 426
               BitDepth: 24
              ColorType: 'truecolor'
        FormatSignature: [137 80 78 71 13 10 26 10]
               Colormap: []
              Histogram: []
          InterlaceType: 'none'
           Transparency: 'none'
 SimpleTransparencyData: []
        BackgroundColor: []
        RenderingIntent: []
         Chromaticities: []
                  Gamma: []
            XResolution: []
            YResolution: []
         ResolutionUnit: []
                XOffset: []
                YOffset: []
             OffsetUnit: []
        SignificantBits: []
           ImageModTime: '1 Mar 2022 18:16:55 +0000'
                  Title: []
                 Author: []
            Description: []
              Copyright: []
           CreationTime: []
               Software: []
             Disclaimer: []
                Warning: []
                 Source: []
                Comment: []
              OtherText: []
```

interaktivni prostredi pro praci s obrazky napriklad mereni vzdalenosti, prohlizeni hodnot jednotlivych obrazku

```
imtool(I);
```



cyklus for

```
for i = 1 : 10
    display(i);
end
```

```
i = 1
i = 2
i = 3
i = 4
i = 5
i = 6
i = 7
i = 8
i = 9
i = 10
```

cyklus while

```
i = 1;
while(i <= 10)
    display(i);
    i = i+1;
end
```

```
i = 1
i = 2
i = 3
i = 4
i = 5
i = 6
i = 7
i = 8
i = 9
i = 10
```

vetveni

```
i = 20
```

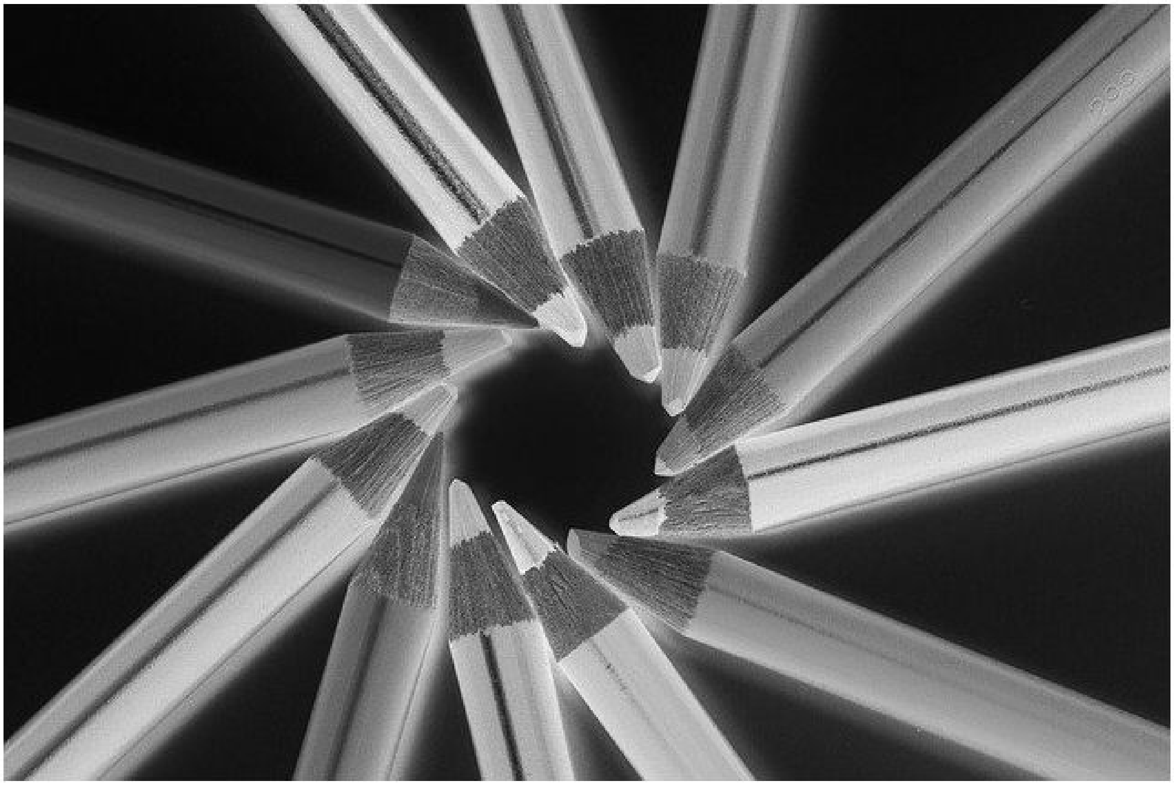
```
i = 20
```

```
if (i > 0)
    display('vetsi');
else
    display('mensi');
end
```

```
vetsi
```

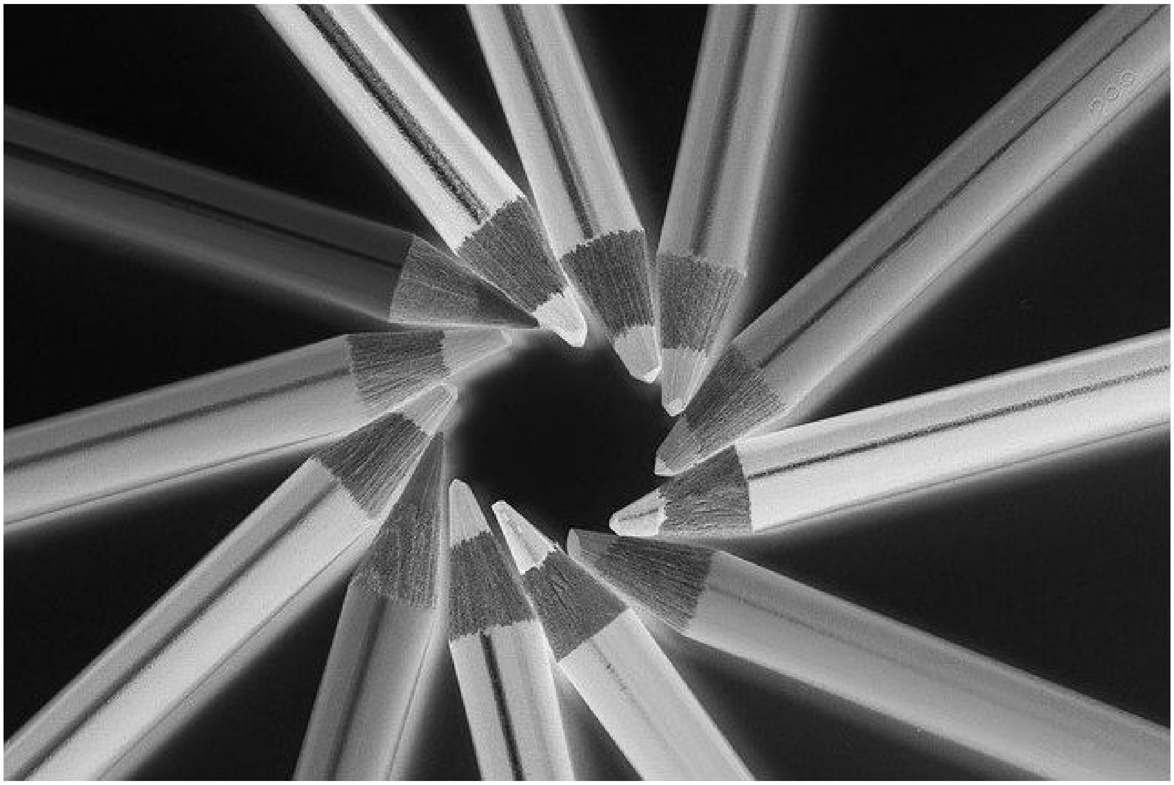
pruchod obrazku pixel po pixelu (2 vnorene cykly)

```
for i = 1:h
    for j = 1:w
        L = 255-I;
    end
end

figure, imshow(L);
```



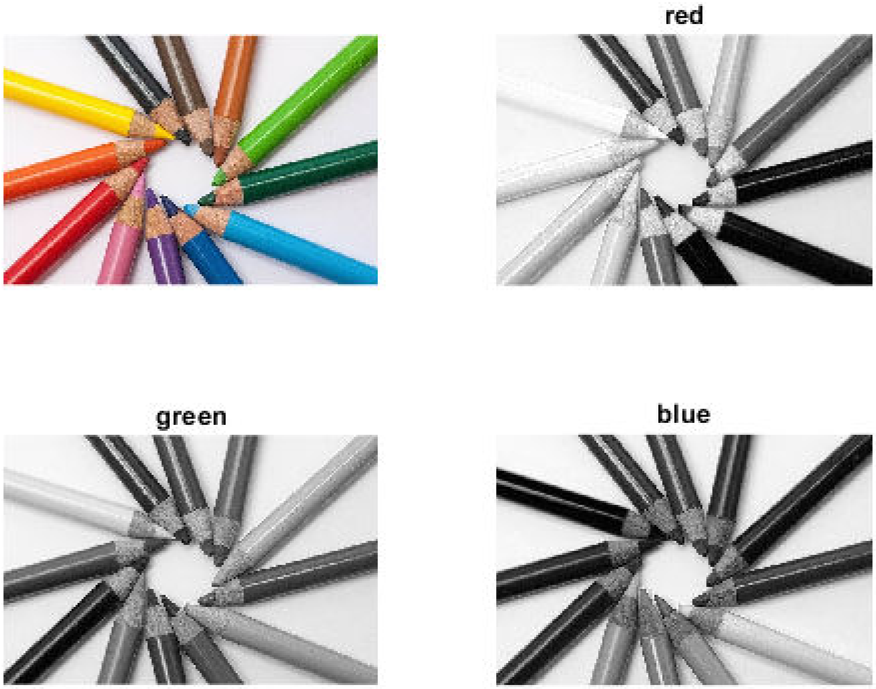
funkce

```
%    function I2 = negativ(I)
%    %NEGATIV Summary of this function goes here
%    %   Detailed explanation goes here
%
%          I2 = 255 - I;
%
%    end

M = negativ(I);
imshow(M);
```



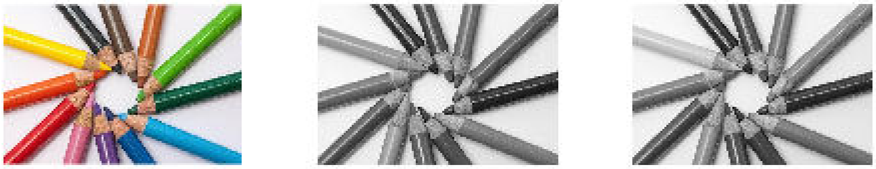
Zobrazeni jednotlivych slozek obrazku

```
figure;
subplot(2,2,1), imshow(I_rgb);
subplot(2,2,2), imshow(I_rgb(:,:,1));
title('red');
subplot(2,2,3), imshow(I_rgb(:,:,2));
title('green');
subplot(2,2,4), imshow(I_rgb(:,:,3));
title('blue');
```



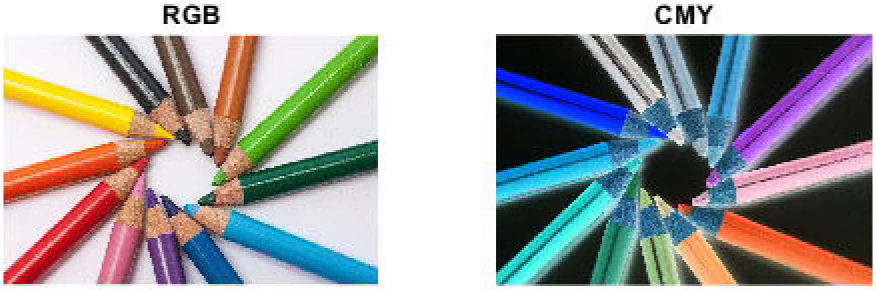
RGB to GRAY

```
figure, subplot(1,3,1), imshow(I_rgb);

% prumerovanim slozek
gray1 = (1/3)*I_rgb(:,:,1) + (1/3)*I_rgb(:,:,2) + (1/3)*I_rgb(:,:,3);
subplot(1,3,2), imshow(gray1,[]);

% vazeny prumer
gray2 = rgb2gray(I_rgb);
subplot(1,3,3), imshow(gray2,[]);
```



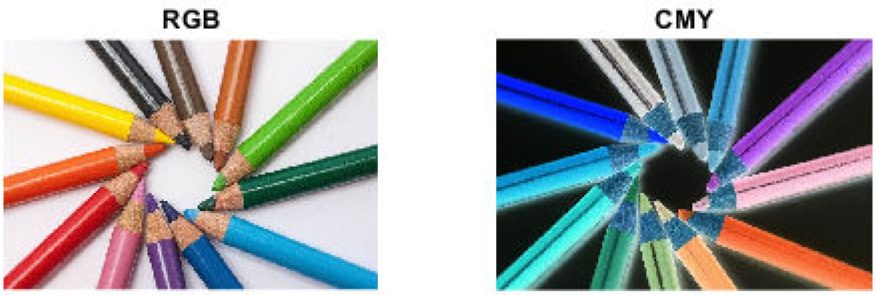
RGB to CMY

```
I_cmy = rgb2cmy(I_rgb);
%I = 255 - I2;
I_rgb2 = cmy2rgb(I_cmy);

figure, subplot(2,2,1), imshow(I_rgb);
title('RGB');
subplot(2,2,2), imshow(I_cmy);
title('CMY');
subplot(2,2,3), imshow(I_rgb2);
title('RGB');
```



CMY to RGB

```
I_rgb = imcomplement(I_cmy);
%I = 255 - I2;

figure, subplot(1,2,1), imshow(I_rgb);
title('RGB');
subplot(1,2,2), imshow(I_cmy);
title('CMY');
```

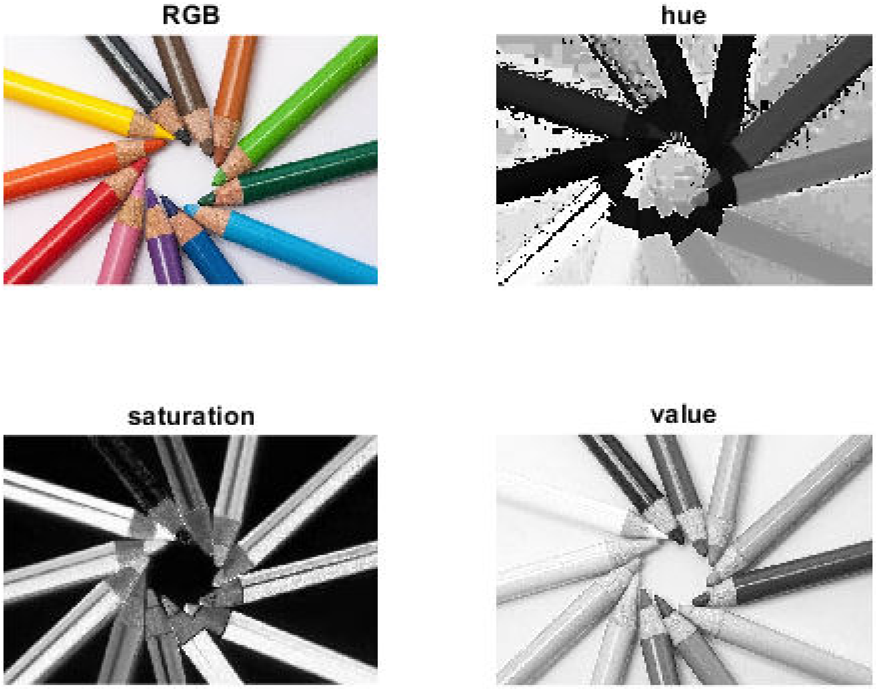


CMYK

```
% vytvoreni tranformace
% pro vice informaci, jake transformace je mozne vytvorit pouzijte:
% help makecform
C = makecform('srgb2cmyk');     %srgb (standard rgb) to cmyk

% aplikace transformace
I_cmyk= applycform(I_rgb,C);
```

RGB to HSV

```
I_hsv = rgb2hsv(I_rgb);

figure, subplot(2,2,1), imshow(I_rgb);
title('RGB');
subplot(2,2,2), imshow(I_hsv(:,:,1));
title('hue');
subplot(2,2,3), imshow(I_hsv(:,:,2));
title('saturation');
subplot(2,2,4), imshow(I_hsv(:,:,3));
title('value');
```

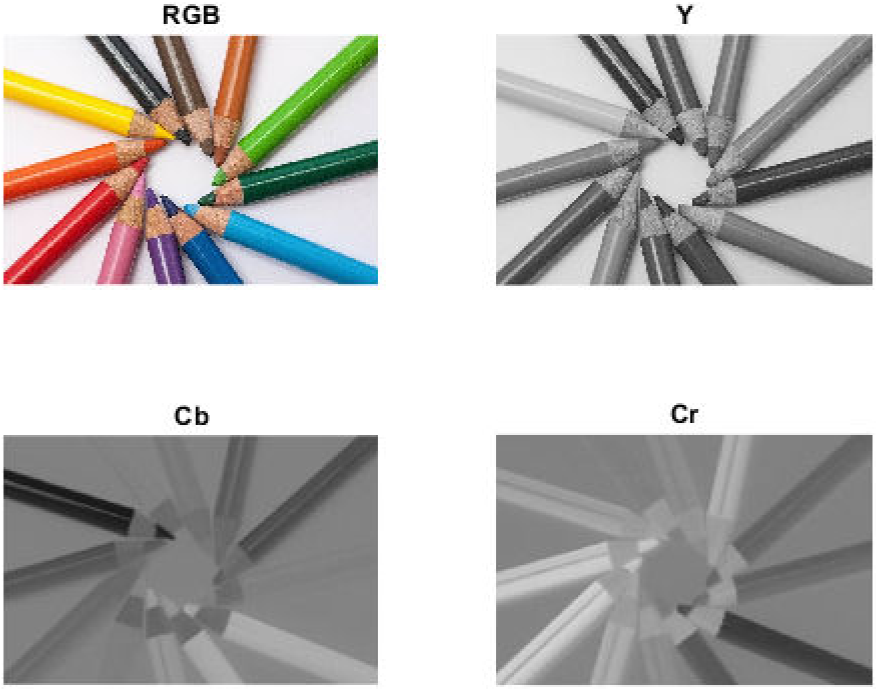


HSV to RGB

```
I_rgb2 = hsv2rgb(I_hsv);

figure, imshow(I_rgb2);
```

RGB to YCbCr

```
I_ycbcr = rgb2ycbcr(I_rgb);

figure, subplot(2,2,1), imshow(I_rgb);
title('RGB');
subplot(2,2,2), imshow(I_ycbcr(:,:,1));
title('Y');
subplot(2,2,3), imshow(I_ycbcr(:,:,2));
title('Cb');
subplot(2,2,4), imshow(I_ycbcr(:,:,3));
title('Cr');
```



YCbCr to RGB

```
I_rgb2 = ycbcr2rgb(I_ycbcr);

figure, imshow(I_rgb2);
```