Homework Assignment #3:

Due: End of Day Nov 23 (Sunday)

Problem 1 Prove the Associativity of Convolution:

$$\big((w * x) * y\big)(m) = \big(w * (x * y)\big)(m)$$

Problem 2

(1) Prove that for an N- dimensional column vector z:

$$\left|(R_k z)^{\wedge}(m)\right| = |\hat{z}(m)|$$

(2) Prove that convolution is also translation invariant:

$$\big((R_k z) * w\big)(m) = \big(R_k(z * w)\big)(m)$$

Problem 3 (Compression)

Perform wavelet compression on the 1D signal below and compare your results to DCT.

In previous HW#2, we have used a step function. In this one, you will use the following 1D Gaussian curve as input and practice compression (1:2, 1:4, 1:8, 1:16) using dwt.

Note that the dwt and idwt function in Matlab only performs 1st stage wavelet transform. How do you perform 2-stage or 3-stage wavelet transform to achieve compression ratio of 1:4 or 1:8?

Matlab code for creating a vector of dimension 1024 of a normal distribution with $\mu = 50$ and $\sigma = 10$:

```
x = 0:0.1:102.3 ;
f = normpdf(x,50, 10);
figure(1)
plot(f,'LineWidth',4) ;
title ('original signal') ;
```

Problem 4 (Wavelet denoising):

Consider the same signal z as in problem 3. Use Matlab functions normrand or wgn to add Gaussian noise to the input signal. Use wavelet transform and Fourier transform to remove the noise and compare the results.

Problem 5 (2D FFT, DCT and Wavelet Transform for Image Compression)

In this problem you will practice with 2D transforms for image compression. Please create your own image and perform compression using 2D transforms such as fft2, dct2, and dwt2.

Hints:

(1) You will need functions imread and imshow to read in an image and display an image.

(2) You may also consider concerting your RGB images into a gray level images first using rgb2gray.

Problem 6 Calculate the gradient and Hessian of the following function f.

$$f(x) = \begin{bmatrix} x_0 & x_1 & x_2 \end{bmatrix} \begin{bmatrix} a_{0,0} & a_{0,1} & a_{0,2} \\ a_{1,0} & a_{1,1} & a_{1,2} \\ a_{2,0} & a_{2,1} & a_{2,2} \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \end{bmatrix}$$

Start with the analytical form of f:

$$f(x) = a_{0,0}x_0^2 + a_{1,1}x_1^2 + a_{2,2}x_2^2 + (a_{0,1} + a_{1,0})x_0x_1 + (a_{0,2} + a_{2,0})x_0x_2 + (a_{1,2} + a_{2,1})x_1x_2$$

and prove that the matrix format of the gradient ∇f and Hessian H_f are correct.

Problem 7 (Implement Team Rank Algorithm)

The following table shows the head-to-head results of the B1G Conference as of Saturday Oct 25, 2025. Note the lower triangular portion of the table is omitted since a game is between two teams. Also, note that the win-loss matrix is NOT symmetric. You can update your matrix by checking https://www.espn.com/college-football/teams and select "Big Ten" when you start working on this problem.

	ILL	IU	IOWA	MD	MSU	MICH	MINN	NEB	NU	ORE	OSU	PSU	PUR	RUTG	UCLA	USC	Wash	WISC
ILL		10-63									16-34		43-27			34-32	25-42	
IU			20-15		38-13					30-20					56-6			
IOWA							41-3					25-24		38-28				37-0
MD								31-34							17-20		20-24	27-10
MSU						20-31		27-38							13-38	31-45		
MICH								30-27								13-31	24-7	24-10
MINN								24-6			3-42		27-20	31-28				
NEB									28-21									
NU										14-34		22-21	19-0		17-14			
ORE																	24-6	34-0
OSU												30-24		56-10				21-7
PSU															37-42			
PUR														24-27				
RUTG																		
UCLA																		
USC																		
WASH																		
WISC																		

Please design a ranking algorithm of B1G schools like the Google Page Rank Algorithm and show your computation results. Does your calculation converge? If it doesn't converge, can you explain why?

How would you adjust your algorithm to accommodate the margin of wins and losses? Does it make any difference to your ranking results?

Can you plot the social network between B1G schools, and see if any clusters emerge?

Hint: you should treat each school as a vertex and if two schools played against each other, introduce an undirected edge between them.