**Homework2**

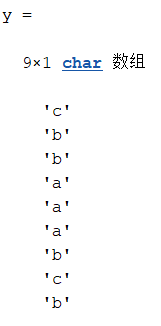
CSCI-6351

Wen Liu

**#Problem1.**

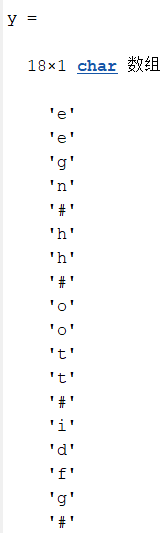
1. For string = “ababbccab”.

The output (y, L) as follow:

y = 

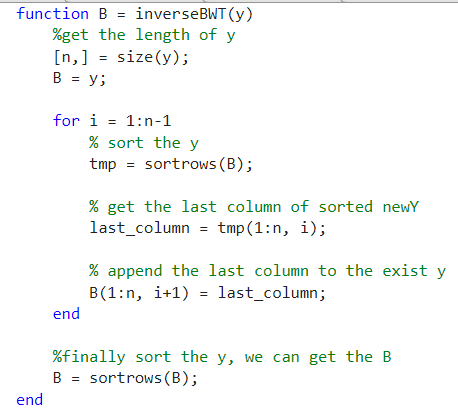
L = 2

For string = “the dog in the fog”. We see # as the white space. The output (y,L) as follow:

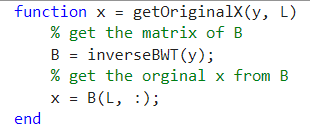
y = 

L = 17

b.



c.



d.

After BWT, we get the (y,L) where y is the last column of B, and L is the location of the original x in B; Therefore we just need to reconstruct the matrix of B from y. For original x, use the B(L) to return the x.

Construct the matrix of B from y: We know the y is the last column of B, and B is the rows of A lexicographically.

Pseudocode code:

B = y

For i = 1:length(y) - 1

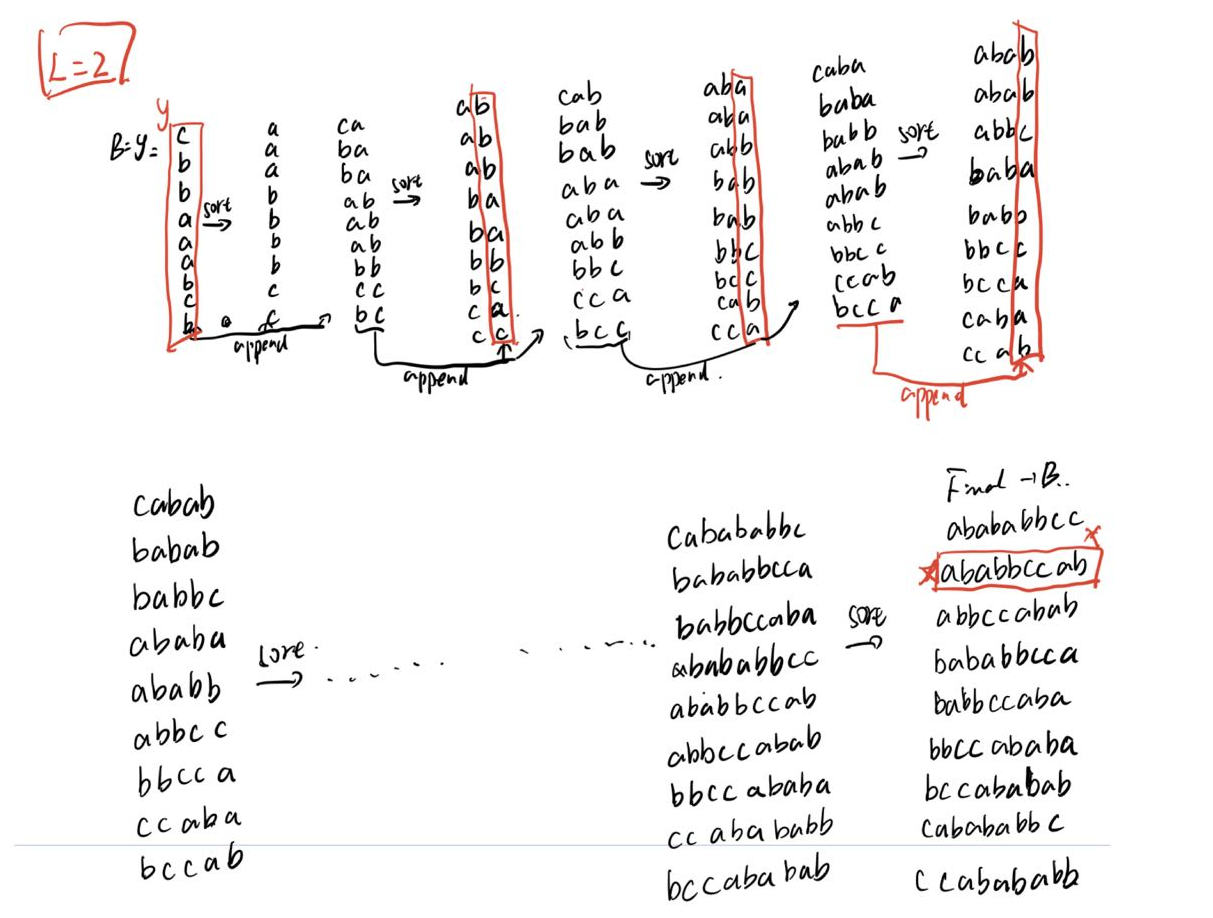
1. Sort the B and get last column of B.

(we use a temporary value to record the sorted B.)

1. append the last column of B to the exist B

B = sort（B）.

End.



**#Problem2.**

All the variables means:

*Xfft* is the Fourier transform of x, *Yfft* is similar.

Xabs is the magnitudes of the elements of Xfft, Yasb is similar

*XHat* is a column derived from *Xfft* by replacing each of the 11 smallest-magnitude elements of *Xfft* by 0, and leaving the other elements intact. *YHat* is defined similarly.

*xHat* is the inverse Fourier transform of *XHat*, *yHat* is similar.

a.

table1 = [Xfft' Xabs' XHat'];

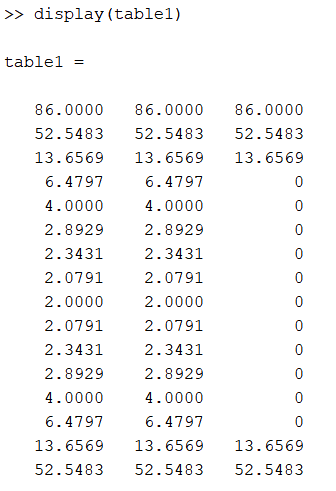


table2 = [Yfft' Yabs' YHat'];

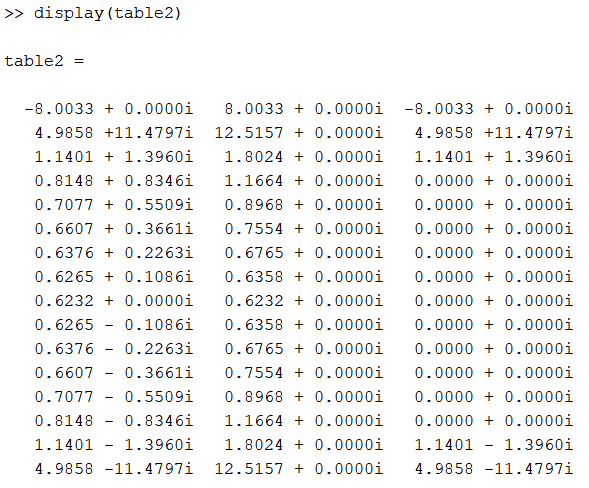


table3 = [x' xHat'];

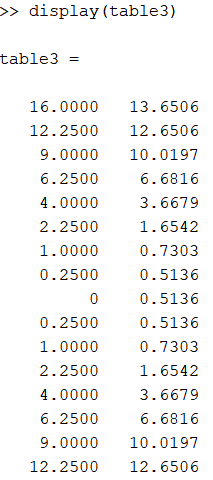
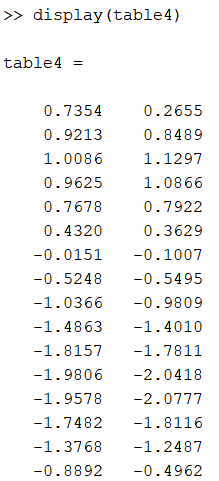
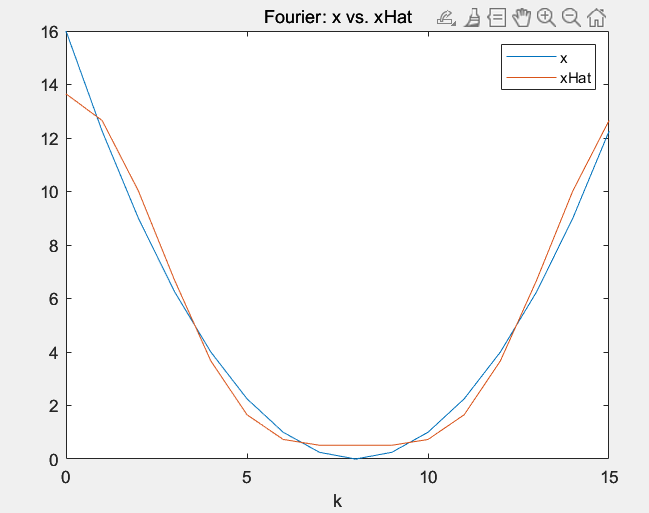
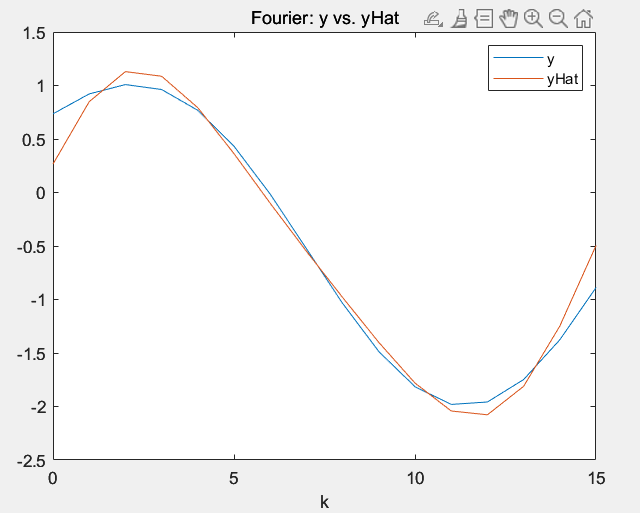


table4 = [y' yHat'];



b.

c.

xMSE = mean((xHat - x).^2) = 0.6107

yMSE = mean((yHat - y).^2) = 0.0296

d.

xSNR = 20 \* log10(x/(x-xHat)) = 17.9494

ySNR = 20 \* log10(y/(y-yHat)) = 12.3148

**#Problem3.**

XDct be the DCT of x, and YDct the DCT of y.

XHatDct be derived from XDct by replacing the last 11 elements of XDct by zeros while keeping the rest of the elements the same YHatDct similarly from YDct.

xHatDct be the inverse DCT of XHatDct, and yHatDct the inverse DCT of YHatDct.

a.

table1 = [XDct' XHatDct']; table2 = [YDct' YHatDct'];

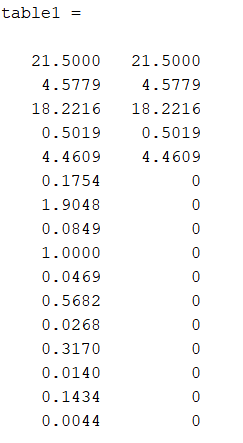
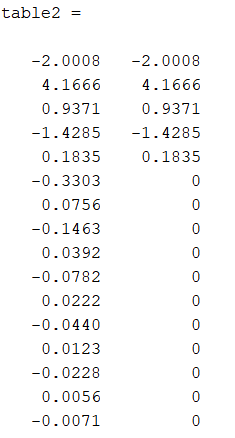
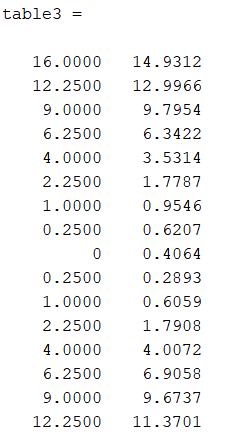
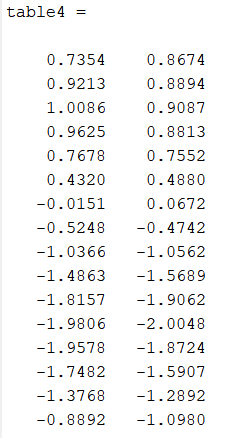
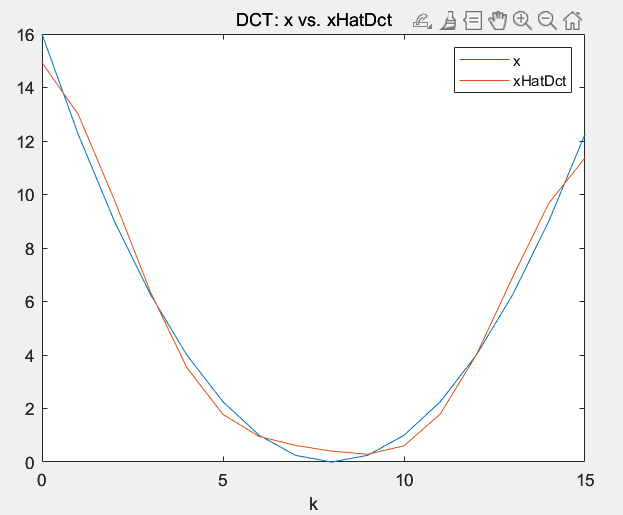
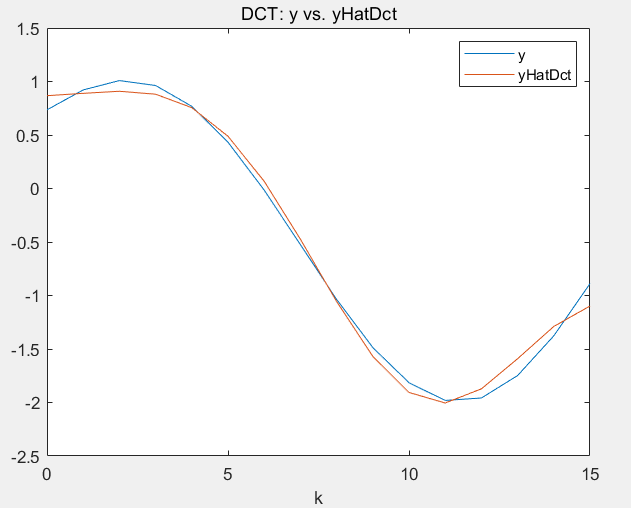
 

table3 = [x' xHatDct']; table4 = [y' yHatDct'];

b.

c.

xMSE = mean((xHatDct - x).^2) = 0.3196

yMSE = mean((yHatDct - y).^2) = 0.0092

d.

xSNR = 20 \* log10(abs(x) / abs(x-xHatDct)) = 21.3106

ySNR = 20 \* log10(abs(y) / abs(y-yHatDct)) = 20.0053

**#Problem4.**

*X\_Hadam* and *Y\_Hadam* are the Hadamard transforms of x and y.

*X\_Hadam\_abs* is the magnitudes of the elements of *X\_Hadam*,

*Y\_Hadam\_abs* is the same.

*X\_Hadam\_Hat* is derived from *X\_Hadam* by replacing the 11 smallest-magnitude elements of *X\_Hadam* by zeros while keeping the rest of the elements the same. *Y\_Hadam\_Hat* is defined similarly.

*smallx\_Hadam\_hat* is the inverse Hadamard transforms of *X\_Hadam\_Hat*,

*smally\_Hadam\_hat* is the same.

a.

table1 = [X\_Hadam X\_Hadam\_abs X\_Hadam\_Hat];

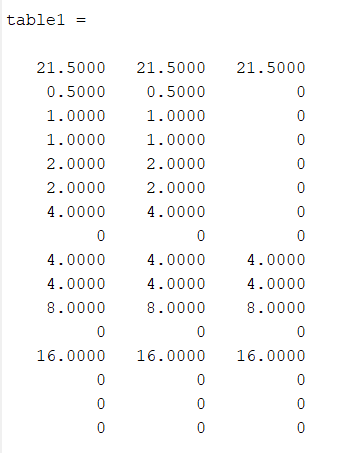


table2 = [Y\_Hadam Y\_Hadam\_abs Y\_Hadam\_Hat];

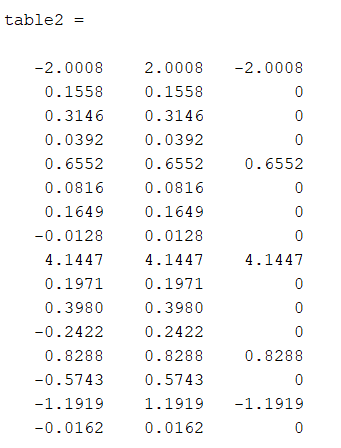


table3 = [x' smallx\_Hadam\_hat];

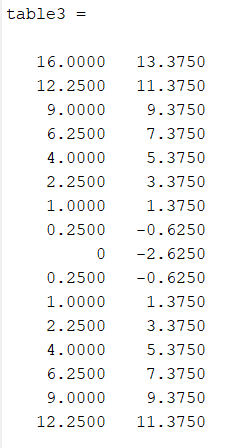
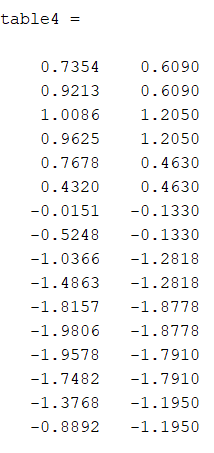


table4 = [y' smally\_Hadam\_hat];

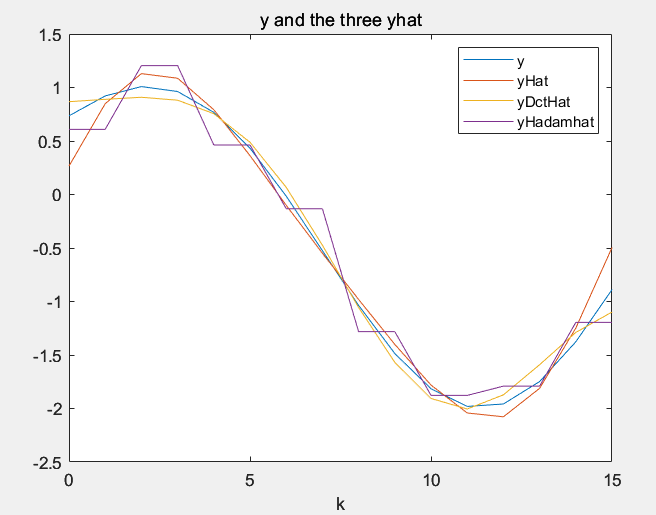
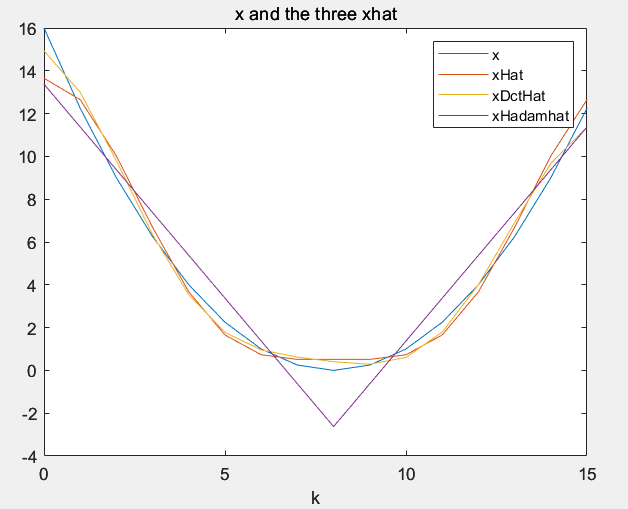


b.

xMSE = mean((smallx\_Hadam\_hat' - x).^2) = 1.6406

yMSE = mean((smally\_Hadam\_hat' - x).^2) = 55.4972

c.



d.



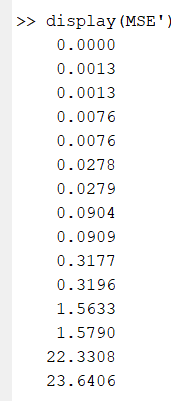
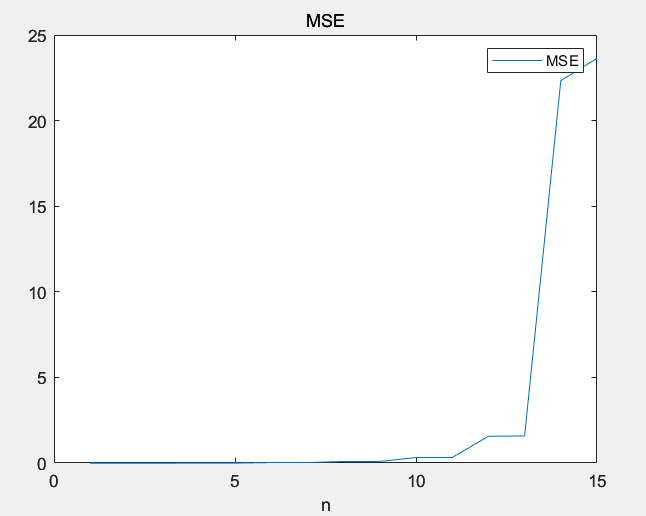


DCT transform gives the best .

DCT transform gives the best .

**#Problem5.**

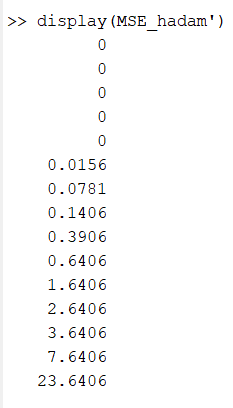
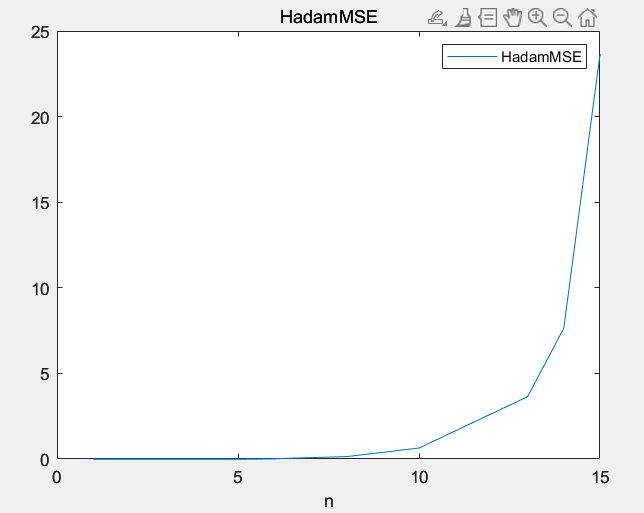
a.

b.

The value of MSE increases as the value of n grows from 1 to 15.

c.

The value of MSE increases as the value of n grows from 1 to 15.