

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΑΤΡΩΝ

ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ : Τ.Μ.Η/Υ.Π

ΕΠΩΝΥΜΟ : ΤΡΙΑΝΤΗΣ

ΟΝΟΜΑ : ΠΑΝΑΓΙΩΤΗΣ

Α.Μ : 5442

ΨΗΦΙΑΚΕΣ ΤΗΛΕΠΙΚΟΙΝΩΝΙΕΣ

ΕΡΓΑΣΤΗΡΙΑΚΗ ΑΣΚΗΣΗ

ΕΡΩΤΗΜΑ 2

Κωδικοποίηση PCM



Αρχικά παρουσιάζονται οι συναρτήσεις που ζητούνται για τον ομοιόμορφο κβαντιστή και για τον ανομοιόμορφο.

```
% my_quantizer : synarthsh tou omoiomorfou kbantisth
% input x,N,min_value,max_value
% output xq,centers

function [xq,centers] = my_quantizer( x,N,min_value,max_value )
%Ypologizoume tis apodektes times toy shmatos me vash tis akraies tou
%shmatos eisodou
x(x>max_value)=max_value; %vriskw th max timh tou input
x(x<min_value)=min_value; %vriskw th min timh tou input
%Ypologizoume to vhma kvantismou
D=(max_value-min_value)/2^(N);
%Ypologizw ta akra tw n diasthmatwn a(1)-a(N+1)
a(1)=min_value;
for i=2:2^(N)+1
a(i)=a(i-1)+D;
end
%Ypologizw ta kentra tw n diasthmatwn
for i=1:2^N
centers(i)=(a(i)+a(i+1))/2;
end
%Antoistixizw kathe deigma sthn perioxh pou anhkei
%kai katalhgw sthn eksodo xq.To
%dianusma auto periexei enan akeraio pou apotelei deikth sto dianusma
%tw n kentrwn kai leei poio kentro (apo ta 2^N)
%antistoixizetai to sugkekrimeno deigma
xlen=max(size(x,1),size(x,2));
%Arxikopoiw to dianusma tw n lusewn
xq=zeros(xlen,1);
for i=1:xlen
flag=true;
k=1;
while(flag)
if(x(i)<=a(k+1))
xq(i)=k;
flag=false;
else
k=k+1;
end
end
end
centers=centers';
end

%% Lloyd_Max : h synarthsh gia ton anomoiomorfo kbantisth
% input x,N,min_value,max_value,Kmax
% output xq,centers,D

function [xq,centers,D] = Lloyd_Max(x,N,min_value,max_value,Kmax)
n = 2 ^ N;
%arxikopoihsh metablhtwn
x1=zeros(size(x));
xq=zeros(size(x));
centers=zeros(n,1);
centers1=zeros(n,1);
T=zeros(n-1,1);
D = zeros(Kmax,1);
```

```

%tropopoihsh tou x wste efarmozei sta oria
for i=1:size(x),
if (x(i) > max_value)
x1(i) = max_value;
elseif (x(i) < min_value)
x1(i) = min_value;
else
x1(i) = x(i);
end
end
%arxikopoihsh tou T
for i = 1 : (n-1),
T(i) = i*(max_value - min_value) / n;
end
T=flipud(T); %ton anapodogyrizoume
%arxikopoihsh kentwn
%an yparxoun x pou anhkoun sto anwtato diasthma vres to kentroeides
if numel(x1(x1<=max_value & x1>T(1))) > 0
centers1(1) = mean(x1(x1<=max_value & x1>T(1)));
%an den uparxoun vale to prwto kentroeides sto meso ths prwths zwnhs
else
centers1(1) = (T(1)+max_value)/2;
end
%an yparxoun x pou anhkoun sto katwtato diasthma vres to kentroeides
if numel(x1(x1<=T(n-1) & x1>=min_value))
centers1(n) = mean(x1(x1<=T(n-1) & x1>=min_value));
%an den uparxoun vale to prwto kentroeides sto meso ths
%teleutaias zwnhs
else
centers1(n) = (T(n-1)+min_value)/2;
end
%gia oles tis upoloipes zwnes kanw to idio
for i = 2 : (n-1)
if numel(x1(x1<=T(i-1) & x1>T(i))) > 0
centers1(i) = mean(x1(x1<=T(i-1) & x1>T(i)));
else
centers1(i) = (T(i-1)+T(i))/2;
end
end
k=2;
D(1) = 0;
D(2) = 1;
%oso h diafora tou distortion apo auto ths prohgomenu iteration
%den exei
%pesei katw apo ena katw orio = 10^-9, epanelave ta vhmata tou
%Lloyd-Max
while k < Kmax && norm( D(k) - D(k-1) ) >= 10^-7
k = k + 1;
centers=centers1;
%epanaupologismos tw n akrwn tw n zwnwn
for i = 1 : (n-1),
T(i) = (centers(i) + centers(i+1)) / 2;
end
%calc xq
xq(x1<=max_value & x1>T(1)) = 1;
xq(x1<=T(n-1) & x1>=min_value) = n;
for i = 2 : (n-1),
xq(x1<=T(i-1) & x1>T(i)) = i;
end
%ypologismos paramorfwsis
D(k) = mean( (x1 - centers(xq)) .^ 2 );

```

```

%% update to akrwtato kentroeides
if numel(x1(x1<=max_value & x1>T(1))) > 0
centers1(1) = mean(x1(x1<=max_value & x1>T(1)));
else
centers1(1) = centers(1);
end
%gia to katw akrwtato
if numel(x1(x1<=T(n-1) & x1>=min_value))
centers1(n) = mean(x1(x1<=T(n-1) & x1>=min_value));
else
centers1(n) = centers(n);
end
%gia ola ta alla
for i = 2 : (n-1),
if numel(x1(x1<=T(i-1) & x1>T(i))) > 0
centers1(i) = mean(x1(x1<=T(i-1) & x1>T(i)));
else
% centers1(i) = centers(i);
end
end
end
D = D';

```

Ζητούμενα Ερωτήματος 2

1) για τη διεκπεραίωση του ερωτήματος δημιουργήθηκαν οι συναρτήσεις :

```

%% SQNR_THEWR H sunarthsh auth upologizei th 8ewrhtikh timh gia to
SQNR
% sthn e3odo tou kvantisth kai epistrefei thn timh auth se db
% OUTPUTS
% sqnr_thewr: H 8ewrhtikh timh tou SQNR sthn e3odo tou kvantisth
function [sqnr_thewr] = sqnr_thewrhtiko(
N,centers,min_value,max_value )
syms x;
P=eval(int(x^2*exp(-x),x,min_value,max_value));
P2=eval(int(x^2*exp(-x),x,max_value,inf));
P=P+P2;
Nq=0;
for i=1:2^N-1
res=int((x-centers(i))^2*exp(-x),min_value+(i-1)*((max_value-
min_value)/2^(N)),min_value+(i)*((max_value-min_value)/2^(N)));
Nq=Nq+eval(res);
end
temp=eval(int((x-centers(length(centers)))^2*exp(-
x),x,max_value,inf));
Nq=Nq+temp;
sqnr_thewr=10*log10(P/Nq);
end

%% SQNR_THEWR H sunarthsh auth upologizei th peiramatikh timh gia to
SQNR
% sthn e3odo tou kvantisth kai epistrefei thn timh auth se db
% OUTPUTS
% sqnr_peir: H peiramatikh timh tou SQNR sthn e3odo tou kvantisth
function [sqnr_peir] = sqnr_peiramatiko( x,xq,centers )
P = mean(x.^2);
xquan = centers(xq); %to kvantismeno shma
Nq = mean((x-xq).^2);
%upologismos tou sqnr se db

```

```
sqr_peir = 10*log10(P/Nq);
end
```

Τα ερωτήματα 1α και 1β απαντιούνται με το εξής script της matlab :

```
% script pou dhmioyrgei thn phgh A, xrhsimopoiwntas ton omoiomorfo
kbantisth
% kwdikopoiiei thn phgh kai ypologizei to SQNR kai oti zhteitai sto 1a
kai
% 1b erwthma

close all; clear all; clc
M = 1e4;

t = (randn(M,1)+sqrt(-1)*randn(M,1))/sqrt(2);
x= abs(t) .^ 2;
% arxikopoihsh pinakwn gia thn apo8hkeush apotelesmatwn
sqr_peir = zeros(2,1);
sqr_thewr = zeros(2,1);
D = zeros(2,1);
% dianysma pou periexei ta diafora N (bits) pou 8eloume na kanoyme th
% kwdikopoihsh ths phghs
K = [4 6];
for i = 1:length(K)
    N = K(i);
    [xq,centers] = my_quantizer( x,N,0,4 );
    [sqr_peir(i)] = sqr_peiramatiko( x,xq,centers );
    display(sqr_peir)
    [sqr_thewr(i)] = sqr_thewrhtiko( N,centers,0,4 );
    display(sqr_thewr)
    D(i) = mean(x-centers(xq)).^2;
    display(D(i))
    figure(i)
    subplot(2,1,1)
    plot(x)
    subplot(2,1,2)
    plot(centers(xq))
end
% arxikopoiw enan counter o opoios metraei poia stoixeia einai ekto
apo ta
% oria 0 kai 4 pou exoume (min_value - max_value)
counter = 0;
for y = 1:length(x)
    if x(y)<0 || x(y)>4
        counter = counter + 1;
    end
end
% edw ypologizetai h peiramatikh pi8anothta yperfortwshs me xrhsh tou
counter
pi8anothta_yperfortwshs_peir = counter/length(x)
```

τα αποτελέσματα που παίρνουμε για κωδικοποίηση σε 4 και 6 bits αντιστοίχα είναι τα εξής :

N =

4

sqrnr_peir =

15.9215

0

sqrnr_thewr =

16.3271

0

D =

1.0e-03 *

0.1823

0

N =

6

sqrnr_peir =

15.9215

16.7222

sqrnr_thewr =

16.3271

17.1996

D =

1.0e-03 *

0.1823

0.2998

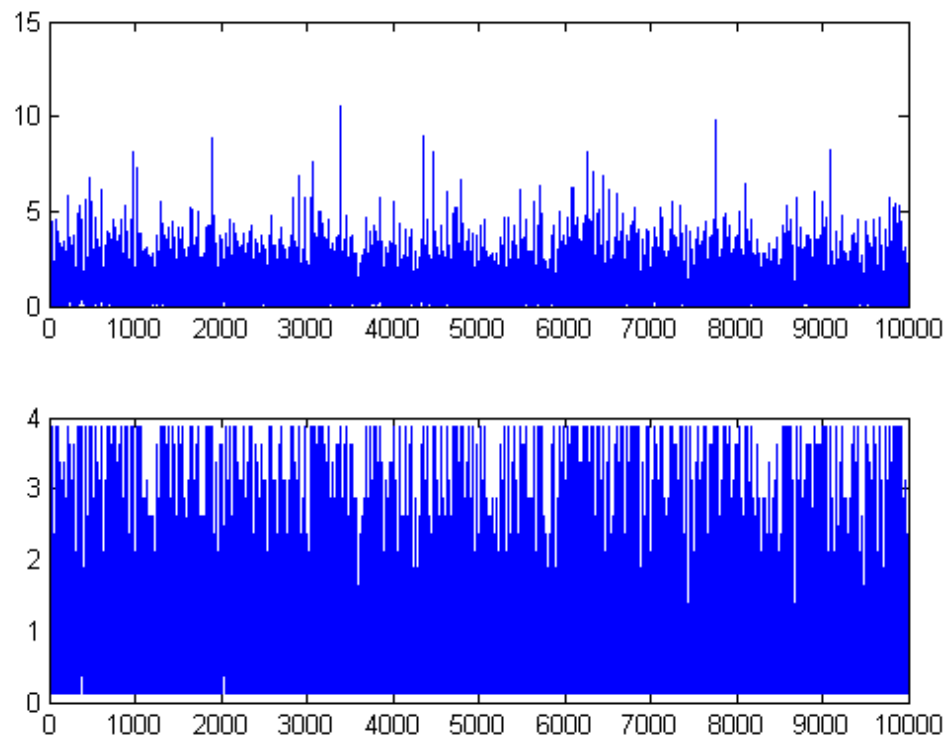
Η πιθανότητα που ζητείται (πειραματική) για το distortion overload είναι :

pi8anothta_yperfortwshs_peir =

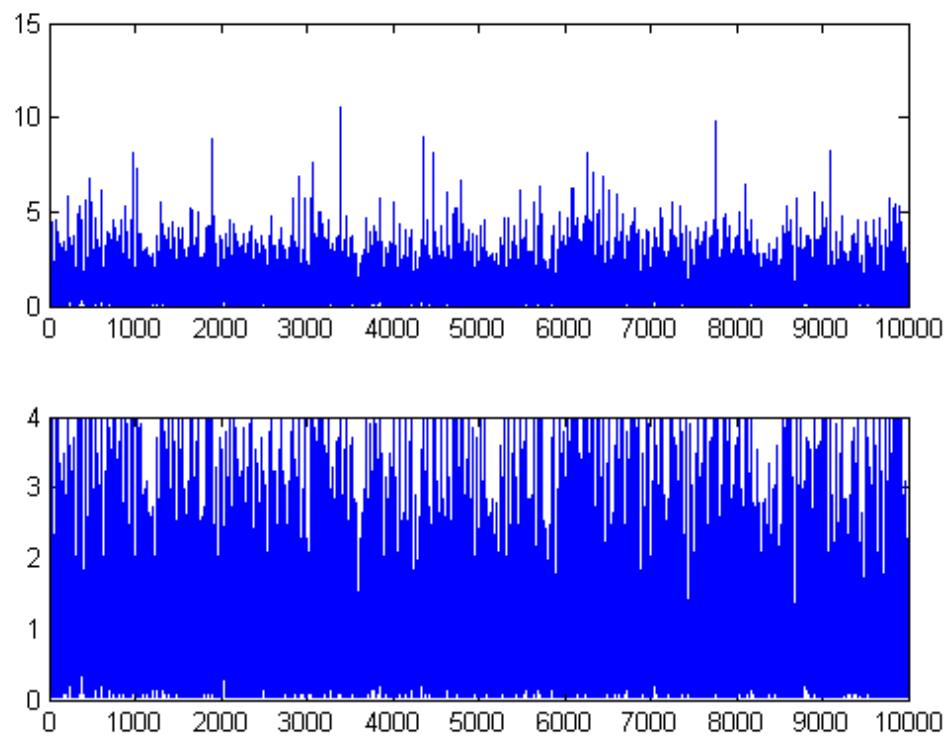
0.0161

Εδώ παρατίθενται 2 γραφικές παραστάσεις που δείχνουν το κανονικό σήμα σε σχέση με το κβαντισμένο

N = 4 bits



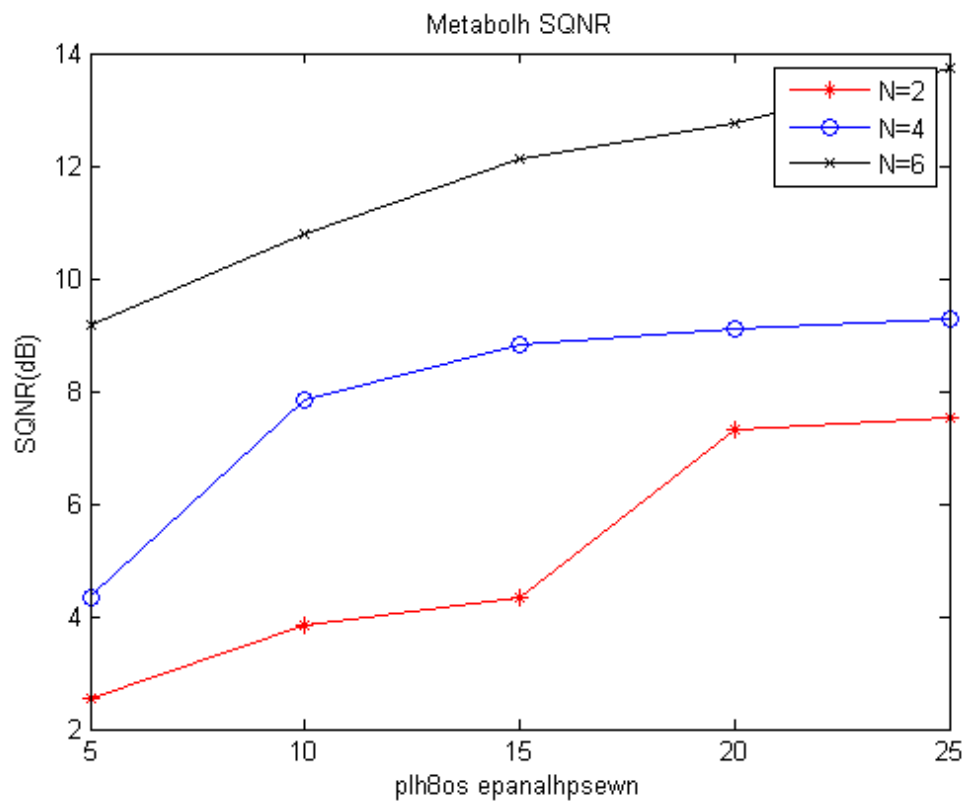
N = 6 bits



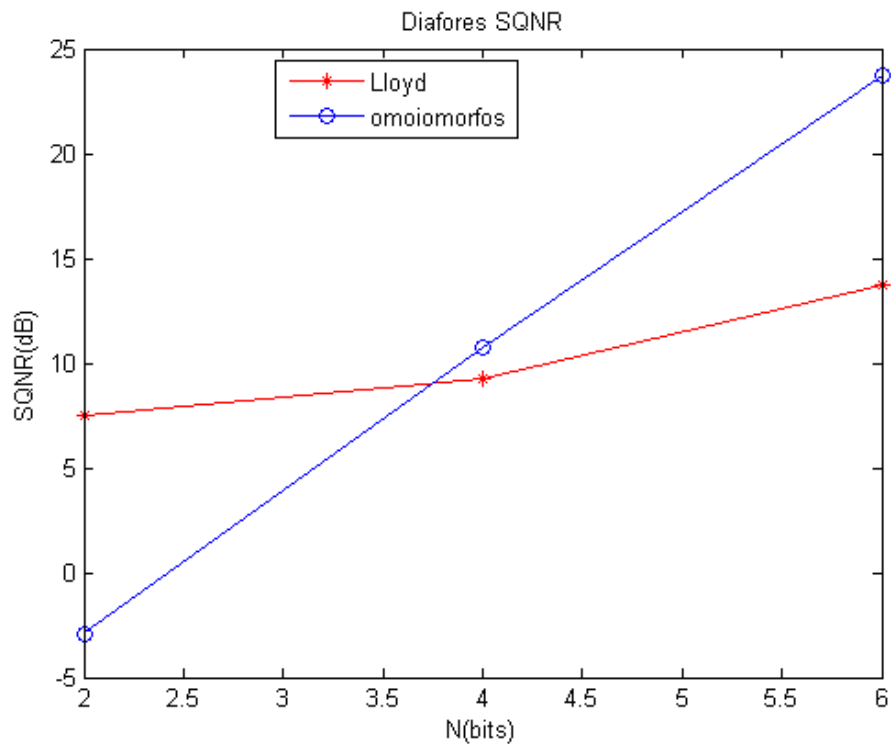
Αν παρατηρήσουμε το σήμα μετά τη κωδικοποίησή του τότε μπορούμε να δούμε ότι για 6 bits είναι καλύτερα κωδικοποιημένο αφού περιλαμβάνει στοιχεία που

είναι ανάμεσα στις τιμές 0-4 και δε τα περιέχει το σήμα που κωδικοποιήθηκε για 4 bits.

2) για το (α) έχουμε τη γραφική



Για το (β) έχουμε τη γραφική που συγκρίνει τα SQNR των δύο κβαντιστών



N =	N =	N =
2	4	6
sq =	sq =	sq =
2.5227	-10.9848	-23.6411
ll_sq =	ll_sq =	ll_sq =
-7.4784	-9.2141	-13.7630

Όπου sq είναι το $10\log_{10}(\text{var}(x - \text{centers}(x_q)))$ & ll_sq είναι το $10\log_{10}(x - \text{Lloyd_centers}(\text{Lloyd_xq}))$. Δηλαδή η διασπορά των διαφορών των κβαντισμένων σημάτων από το αρχικό προς τη διασπορά του αρχικού σήματος. Έτσι βλέπουμε κατά πόσο «χάνει» από το αρχικό σήμα. Για 2 bits κωδικοποίηση ο ομοιόμορφος κωδικοποιητής κωδικοποιεί ικανοποιητικά ενώ μετά πιο ικανοποιητική είναι η κωδικοποίηση του μη ομοιόμορφου κωδικοποιητή.

Ο κώδικας που χρησιμοποιήθηκε για το 2α κ 2β είναι :

```
%% script gia th phgh B kai kalypetei ta erwthmata 2a - 2b

close all; clear all; clc

% arxikopoihsh pinakwn gia thn apo8hkeush apotelesmatwn
sqnr_omoiomorfou_peir = zeros(3,1);
Lloyd_sqnr_peir = zeros(3,5);
apodosh_om = zeros(3,1);
apodosh_anom = zeros(3,1);
apd_ws_pros_arxiko = zeros(3,1);
apd_ws_pros_arxiko_Lloyd = zeros(3,1);
% xrhsimopoioume thn audioread epeidh etsi epideiknyei h matlab
[x,fs] = audioread('speech.wav');
% dianysma pou perixe ta diafora N (bits) pou 8eloume na kanoyme th
% kwidikopoihsh ths phghs
K = [2 4 6];
for i = 1:length(K)
    N = K(i);
    [xq,centers] = my_quantizer(x,N,-1,1);
    sqnr_omoiomorfou_peir(i,1) = sqnr_peiramatiko(x,xq,centers);
    kmax = 5:5:25;
    for j = 1:length(kmax)
        [Lloyd_xq,Lloyd_centers,D] = Lloyd_Max(x,N,-1,1,kmax(j));
        Lloyd_sqnr_peir(i,j) =
sqnr_peiramatiko(x,Lloyd_xq,Lloyd_centers);
    end
    display(N)
    sq = 10*log10(var(x-centers(xq))/var(x))
    apd_ws_pros_arxiko(i) = sq;
    ll_sq = 10*log10(var(x-Lloyd_centers(Lloyd_xq))/var(x))
    apd_ws_pros_arxiko_Lloyd(i) = ll_sq;
end
figure(1)
plot(kmax,Lloyd_sqnr_peir(1,:), 'r-*)
hold on
```

```

plot(kmax,Lloyd_sqnr_peir(2,:), 'b-o')
hold on
plot(kmax,Lloyd_sqnr_peir(3,:), 'k-x')
hold off
legend('N=2', 'N=4', 'N=6')
title('Metabolh SQNR')
xlabel('plh8os epanalhpsewn')
ylabel('SQNR(dB)')

figure(2)
N = 2:2:6;
plot(N,Lloyd_sqnr_peir(:,5), 'r-*)
hold on
plot(N,sqnr_omoiomorfou_peir(:,:), 'b-o')
hold off
legend('Lloyd', 'omoiomorfos')
title('Diafores SQNR')
xlabel('N(bits)')
ylabel('SQNR(dB)')

```

2γ) η συνάρτηση που ζητείται είναι η :

```

%% ypologismos entropias kai pi9anothtas emfanishs ka8e epipedou
kbantishs
% input x,xq,centers
% x : arxiko shma, xq : kbantismeno shma, centers : ypologismena
kentra
% output level_entropy,level_prob,entropy
% level_entropy : entropia ka8e epipedou kbantishs
% level_prob : pithanothta emfanishs ka8e epipedou
% entropy : h synolikh entropia
function [level_entropy,level_prob,entropy] =
erotima_2c(x,xq,centers)

```

```

y = unique(centers(xq));
count = zeros(length(y),1);
for i = 2:length(y)
    counter = 0;
    for j = 1:length(x)
        if y(i-1,1)<=x(j) && x(j)<=y(i,1)
            counter = counter + 1;
        end
        count(i-1) = counter;
        if i == length(y)
            count(i) = counter;
        end
    end
end

level_prob = count./sum(count);
entropy = 0;
for i = 1:length(level_prob)
    if level_prob(i)~=0
        entropy = entropy + (-level_prob(i)*log2(level_prob(i)));
    end
end

level_entropy = zeros(length(y),1);
for i = 1:length(level_prob)
    if level_prob(i)~=0
        level_entropy(i) = -level_prob(i)*log2(level_prob(i));
    end
end

```

```

end
end
end

```

και το script που απανταει στο ερωτημα αυτο είναι :

```

%% script poy apantaei sto erotima 2c
% ypologizetai h pi8anothta emfanishs ka8e epipedou kai apo8hkeuetai
se
% cell, akomh ypologizetai h entropia ka8e epipedou kbantishs kai
% apo8hkeutai se cell kai ypologizw kai thn entropia gia N = 2:2:6
gia ka8e
% kbantismeno shma dhladh kai apo to shma pou bgazei san e3odo o
% omoiomorfos kbantisths kai o anomoiomorfos
close all; clear all; clc

sqnr_omoiomorfou_peir = zeros(3,1);
Lloyd_sqnr_peir = zeros(3,5);
[x,fs] = audioread('speech.wav');
level_entropy_omoio = cell([3,1]);
level_prob_omoio = cell([3,1]);
level_entropy_anomoio = cell([3,1]);
level_prob_anomoio = cell([3,1]);
entropy_omoio = zeros(3,1);
entropy_anomoio = zeros(3,1);
K = [2 4 6];
for i = 1:length(K)
    N = K(i);
    [xq,centers] = my_quantizer(x,N,-1,1);
    sqnr_omoiomorfou_peir(i,1) = sqnr_peiramatiko(x,xq,centers);
    [level_entropy_om,level_prob_om,entropy_om] =
erotima_2c(x,xq,centers);
    level_entropy_omoio{i,1} = level_entropy_om;
    level_prob_omoio{i,1} = level_prob_om;
    entropy_omoio(i,1) = entropy_om;
    kmax = 5:5:25;
    for j = 1:length(kmax)
        [Lloyd_xq,Lloyd_centers,D] = Lloyd_Max(x,N,-1,1,kmax(j));
        Lloyd_sqnr_peir(i,j) =
sqnr_peiramatiko(x,Lloyd_xq,Lloyd_centers);
        [level_entropy_anom,level_prob_anom,entropy_anom] =
erotima_2c(x,Lloyd_xq,Lloyd_centers);
        level_entropy_anomoio{i,1} = level_entropy_anom;
        level_prob_anomoio{i,1}= level_prob_anom;
        entropy_anomoio(i,1) = entropy_anom;
    end
end
end

for i = 1:3
    figure(i)
    plot(level_prob_omoio{i,1}, 'b-*)
    hold on
    plot(level_prob_anomoio{i,1}, 'r->')
    hold off
end

```

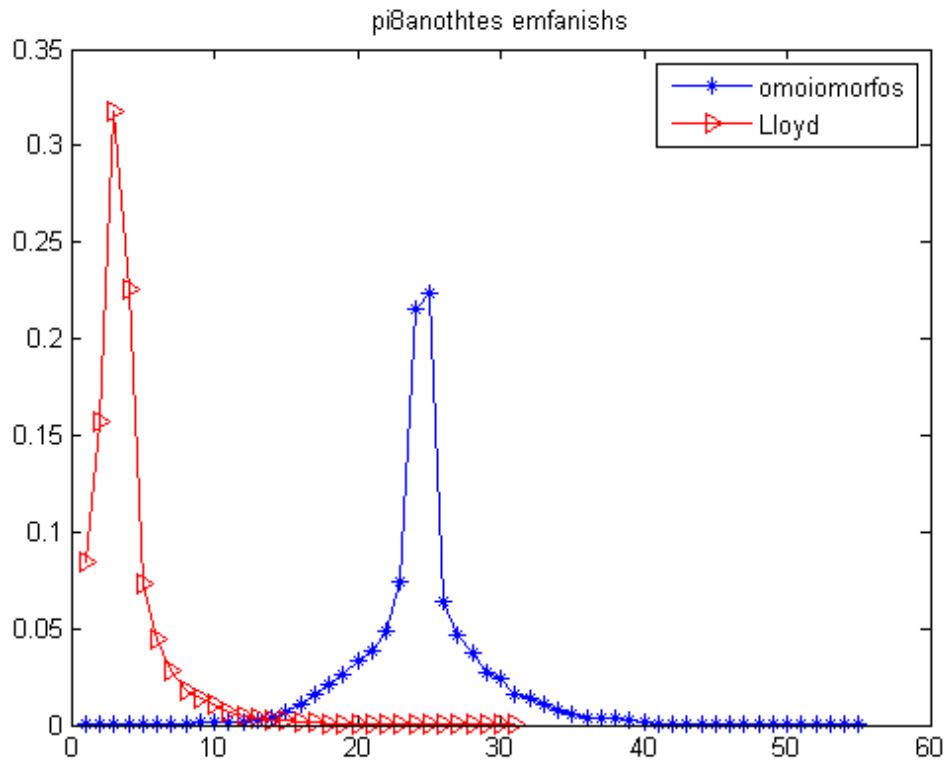
πιθανότητες ομοιομορφου

			0,08416	
			0,15746	
			0,3182	
			0,22544	
			0,07343	
			0,04425	
			0,02784	
			0,01841	
			0,01354	
			0,01035	
			0,00682	
			0,00454	
			0,00317	
			0,00287	
			0,00246	
			0,00188	
			0,00109	
			0,00099	
			0,00053	
			0,00066	
			0,00041	
			0,00025	
		0,29947	0,0002	
		0,48498	0,00018	
		0,12249	0,00015	
		0,05763	5,07E-05	
		0,02342	0,00015	
	0,36818	0,00816	0,00013	
	0,46063	0,00227	0,00015	
	0,0856	0,00079	0,00013	
N = 2	0,0856	N=4	0,00079	
			N=6	0,00013

Οι εντροπιες για $N = 2:2:6$ για τον ομοιομορφο και τον ανομοιομορφο είναι :

N(bits)	Εντροπία ομοιόμορφου	Εντροπία ανομοιόμορφου
2	0.6998	1.6530
4	1.9760	1.8553
6	3.8264	2.8388

Και σε γραφικες για τις πιθανοτητες παιρνουμε



2δ)

Για να παρουμε την τιμη του mse

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^n (\hat{Y}_i - Y_i)^2$$

Χρησιμοποιήσαμε με βάση τον τύπο τον κώδικα :

```
mse_omoiom= sum((centers(xq)-x).^2)/length(x)
mse_Lloyd= sum((Lloyd_centers(Lloyd_xq)-x).^2)/length(x)
```

******(ο κώδικας αυτός προστέθηκε στο erotima 2c script.m στο loop της for και βγαζει στο command window σε κάθε loop το αποτέλεσμα του mse κάθε φορά και πήραμε τα αποτελέσματα)******

N(bits)	mse(omoiomorfou)	Mse(Lloyd)
2	0.0362	0.0033
4	0.0016	0.0022
6	7.9168e-05	7.8513e-04