Μάθημα: «Αρχές Εφαρμογές Σημάτων και Συστημάτων (4ο Εξάμηνο)»

ΟΜΑΔΑ ΕΡΓΑΣΙΑΣ:

Αναστάσιος Κωνσταντινίδης Π17057
Φώτιος Σιούζιος Π17121
Χρήστος Δήμτσας Π17027

Γ1

1.

Με Βάση το σήμα $x(t) = \cos(100\pi t) + \cos(200\pi t) + \sin(500\pi t)$

Και τον τύπο $T = 2\pi/\omega$.

Έχουμε, πως το $\cos(100\pi t)$ έχει $T1 = 0{,}02sec$

Το $\cos(200\pi t)$ έχει $T2 = 0{,}01sec$

Το $\sin(500\pi t)$ έχει $T3 = 0{,}004sec$

και η περίοδος του σήματος ισούται με $Ts = 2\pi/\omega = 0{,}02$ sec.

Άπο το θεώρημα Shannon – Nyquist έχουμε : $1/T \leq 2s(o)$.

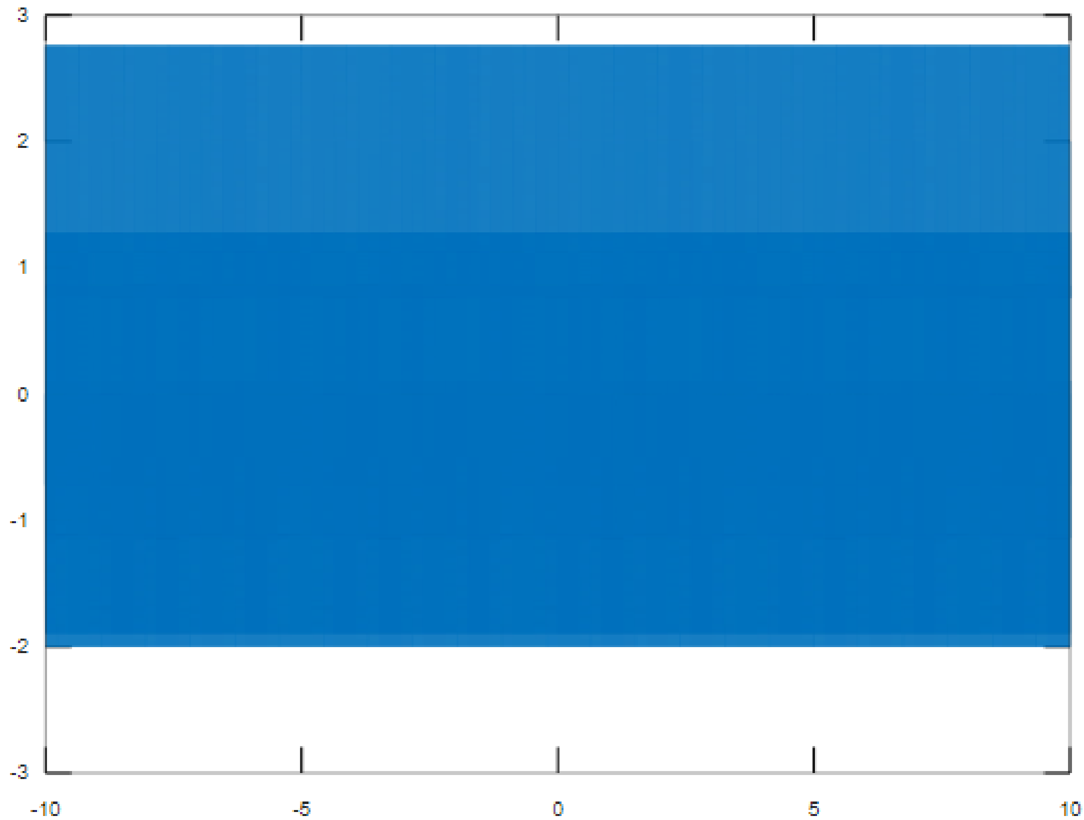
Άρα: $s(o) = 1/2T = 1/0{,}04 = 25$ Hz

Η ελάχιστη απαιτούμενη συχνότητα θα είναι $s(o) = 25$ Hz

2.

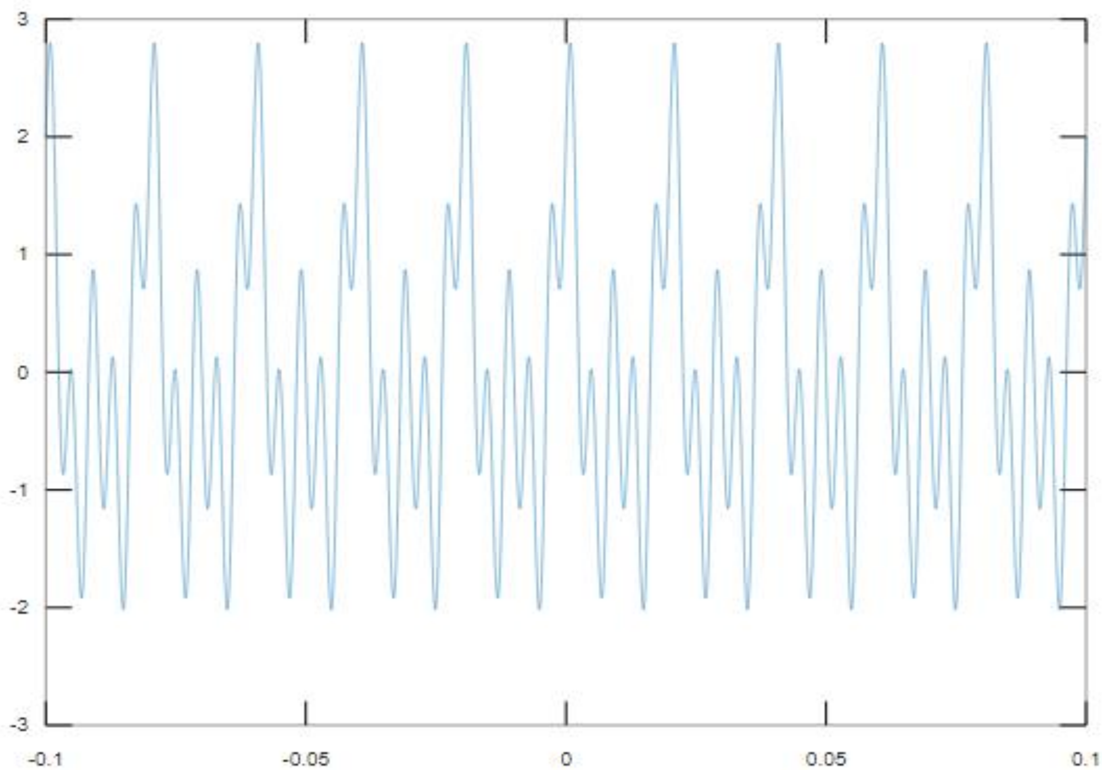
Έχουμε x(t) όπου t ανήκει [-10,10],με βήμα Δt=0,001s

```
octave:2> t=-10:0.001:10;
octave:3> x= cos(100*pi*t) + cos(200*pi*t) + sin(500*pi*t);
octave:4> plot(t,x);
```



Μειώνουμε την κλίμακα, με το να διαιρέσουμε με 100 τον χρόνο για να φανεί καλύτερα το σχήμα.

```
octave:2> t=-0.1:0.00001:0.1;
 x= cos(100*pi*t) + cos(200*pi*t) + sin(500*pi*t);
plot(t,x);
```



3.

Με συχνότητα δειγματοληψίας S(o)=25Hz έχουμε ότι :

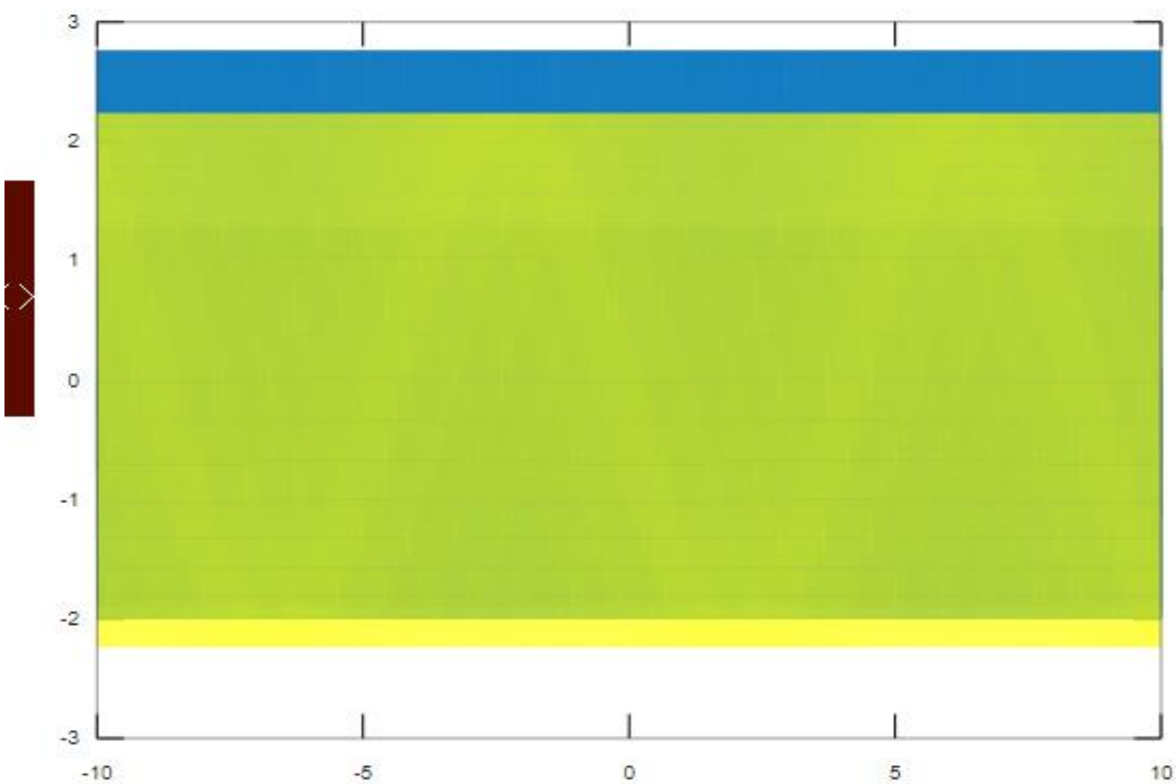
T=0,04sec

T=2π/ω ⇔ ω=2π/T ⇔ ω=2π/0,04 ⇔ <u>ω=50π</u>

Έτσι λοιπόν το ανακατασκευασμένο σήμα είναι το ακόλουθο :

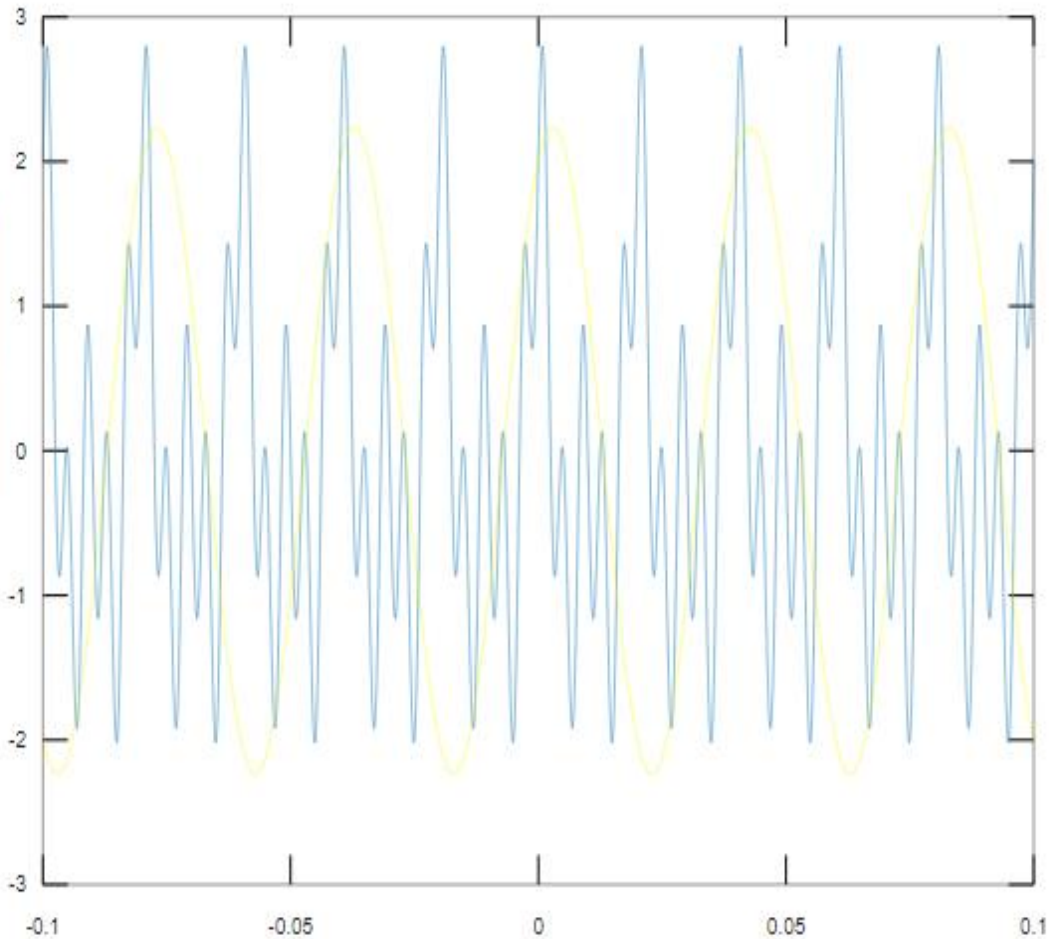
y(t)=cos(50πt)+cos(50πt)+sin(50πt)

```
octave:15>
 t=-10:0.001:10;
 x= cos(100*pi*t) + cos(200*pi*t) + sin(500*pi*t);
 y=cos(50*pi*t)+cos(50*pi*t)+sin(50*pi*t);
 plot(t,x,t,y,'y');
```



Μειώνουμε την κλίμακα, με το να διαιρέσουμε με 100 τον χρόνο για να φανεί καλύτερα το σχήμα.

```
octave:5> t=-0.1:0.00001:0.1;
 x= cos(100*pi*t) + cos(200*pi*t) + sin(500*pi*t);
 y=cos(50*pi*t)+cos(50*pi*t)+sin(50*pi*t);
 plot(t,x,t,y,'y');
```



4.

Έστω νέα συχνότητα δειγματοληψίας είναι

S2 = 40 Hz.

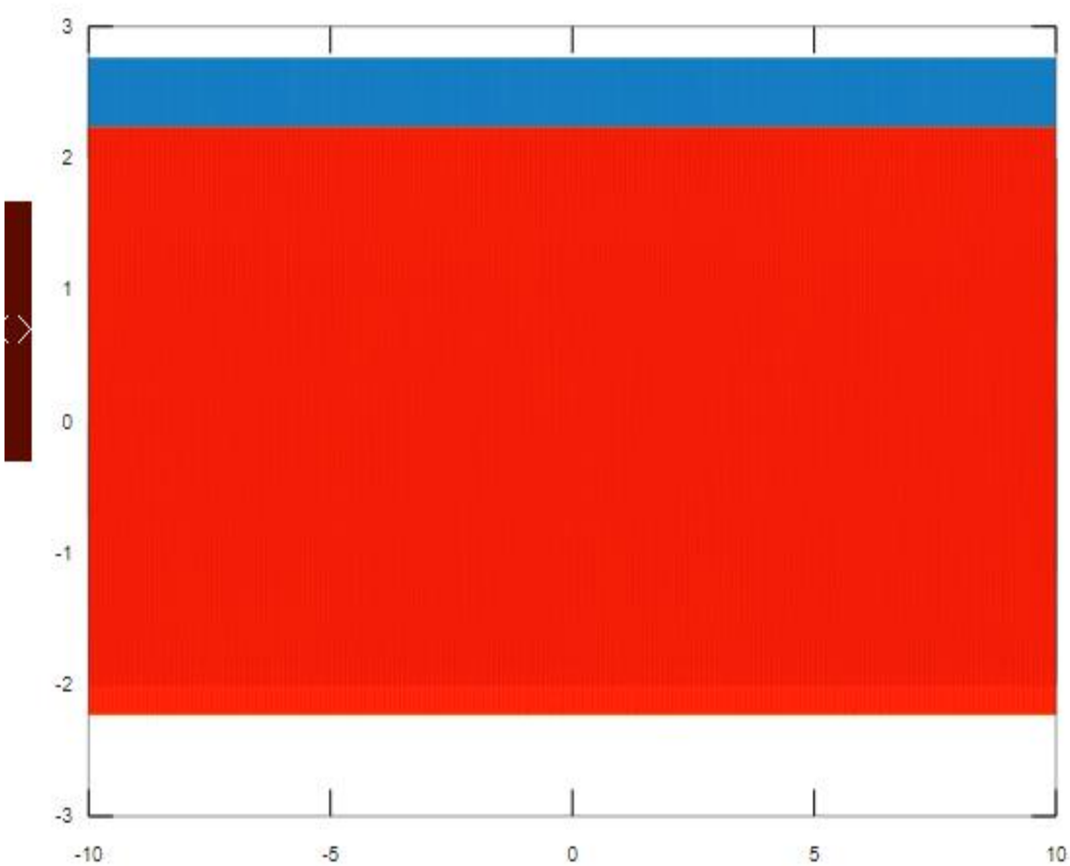
Επομένως T2 = 1/S2 = 0,025 sec και ω2 = 2π/T2 = 80 π.

Άρα το καινούργιο σήμα είναι το

y2(t) = cos(80*pi*t) + cos(80*pi*t) + sin(80*pi*t);

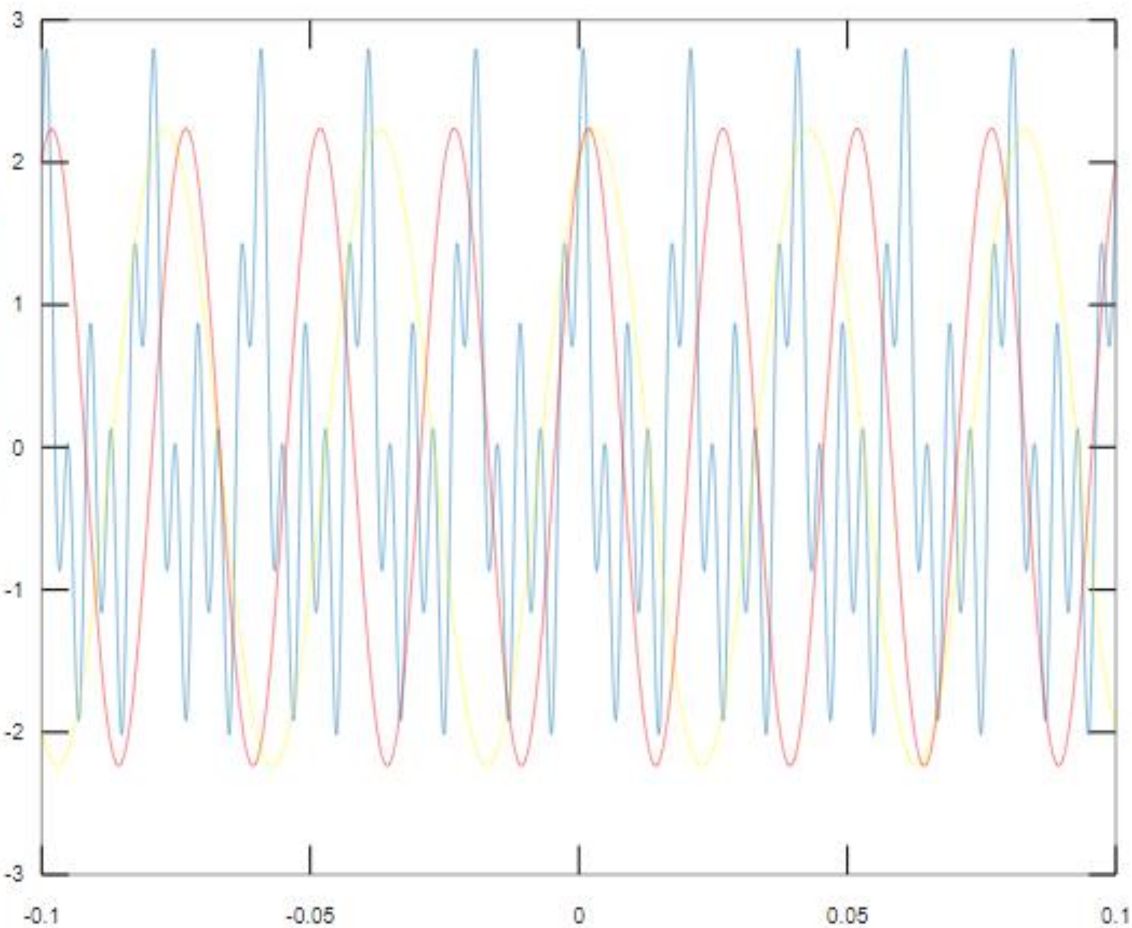
και έτσι προκύπτει το παρακάτω σχήμα.

```
octave:24>  t=-10:0.001:10;
 x= cos(100*pi*t) + cos(200*pi*t) + sin(500*pi*t);
 y=cos(50*pi*t)+cos(50*pi*t)+sin(50*pi*t);
 y2= cos(80*pi*t) + cos(80*pi*t) + sin(80*pi*t);
 plot(t,x,t,y,'y',t,y2,'r');
```



Μειώνουμε την κλίμακα, με το να διαιρέσουμε με 100 τον χρόνο για να φανεί καλύτερα το σχήμα.

```
octave:9> t=-0.1:0.00001:0.1;
 x= cos(100*pi*t) + cos(200*pi*t) + sin(500*pi*t);
 y=cos(50*pi*t)+cos(50*pi*t)+sin(50*pi*t);
 y2= cos(80*pi*t) + cos(80*pi*t) + sin(80*pi*t);
 plot(t,x,t,y,'y',t,y2,'r');
```



5.

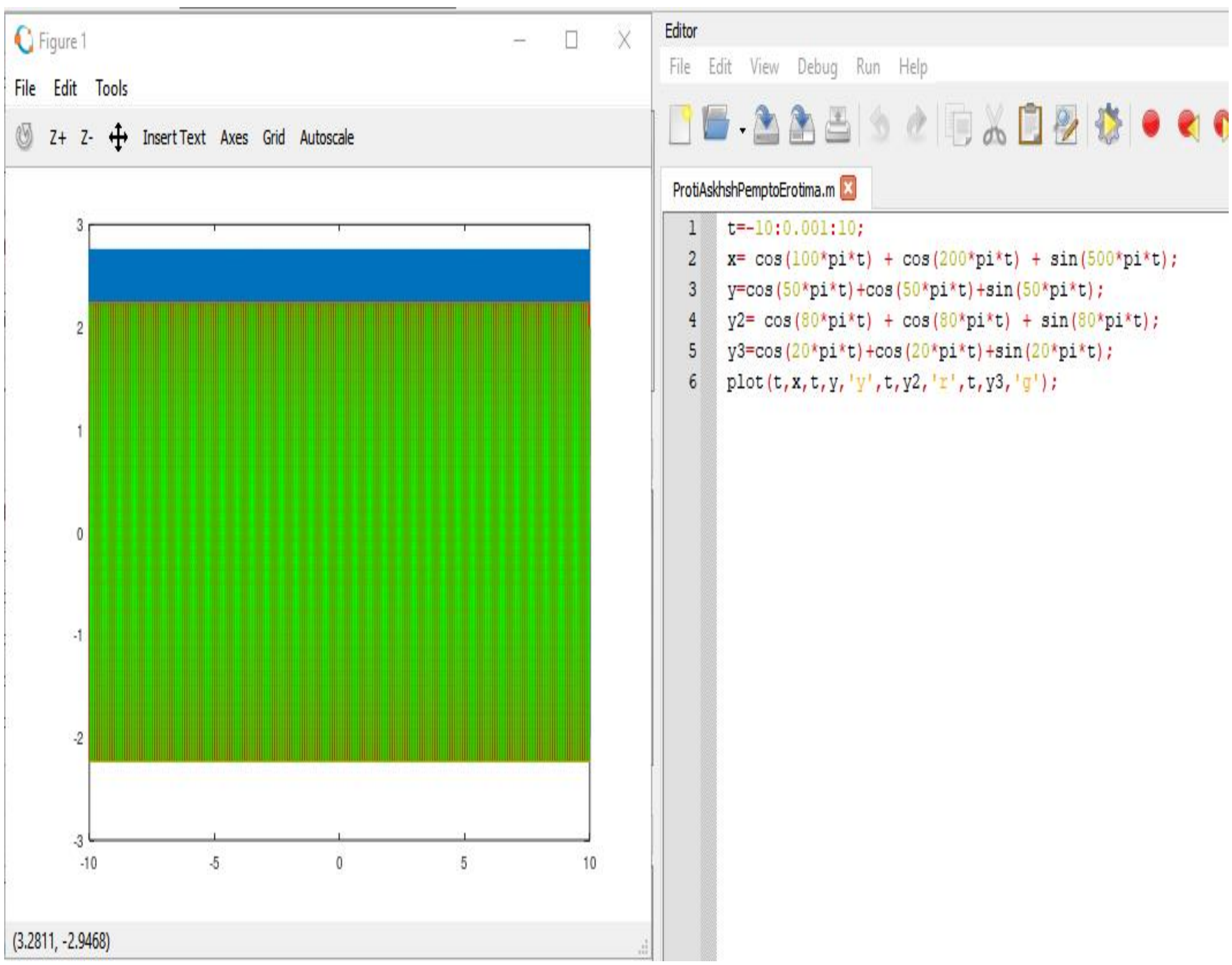
Επιλέξαμε συχνότητα δειγματοληψίας μικρότερη από 25Hz. Θεωρούμε έτσι **S3=10Hz.**

$S3=1/T \Leftrightarrow T=1/S3 \Leftrightarrow T=0{,}1sec$

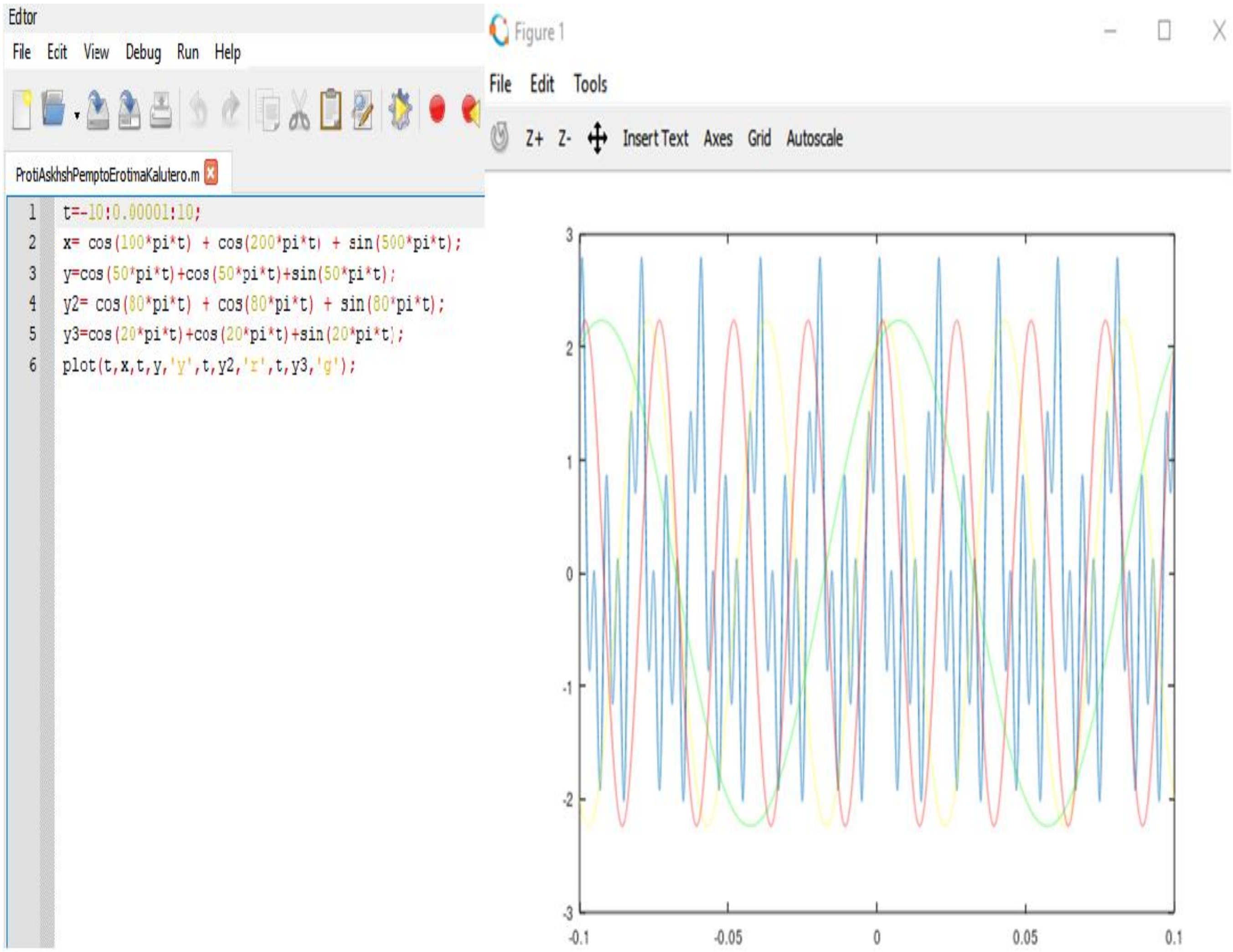
$T=2\pi/\omega \Leftrightarrow \omega=2\pi/T \Leftrightarrow \omega=2\pi/0{,}1 \Leftrightarrow \omega=20\pi$

Συνεπώς το νέο σήμα είναι :

y3(t)=cos(20πt)+cos(20πt)+sin(20πt)



Μειώνουμε την κλίμακα, με το να διαιρέσουμε με 100 τον χρόνο για να φανεί καλύτερα το σχήμα.



6.

Παρατηρούμε ότι από τα τέσσερα σήματα το σήμα **y3(t)** λόγω του ότι έχει τη μικρότερη συχνότητα(**f=10Hz**) έχει και το μεγαλύτερο μήκος κύματος ενώ το σήμα **y2(t) π**ου έχει την μεγαλύτερη συχνότητα έχει το μικρότερο μήκος κύματος. Επίσης διακρίνουμε ότι, το σήμα **y2(t) ,** έχει μικρότερη απώλεια πληροφορίας από το σήμα **y3(t)** σε σύγκριση με το αρχικό σήμα **x(t)** και αυτό οφείλεται στην μεγαλύτερη συχνότητα που επιλέξαμε. **Επιπλέον μπορούμε να πούμε ότι το πρώτο σήμα(το ανακατασκευασμένο) y(t) που ικανοποιεί το θεώρημα δειγματοληψίας κρατάει αναλλοίωτη(πιστή) την πληροφορία του αρχικού σήματος x(t).**

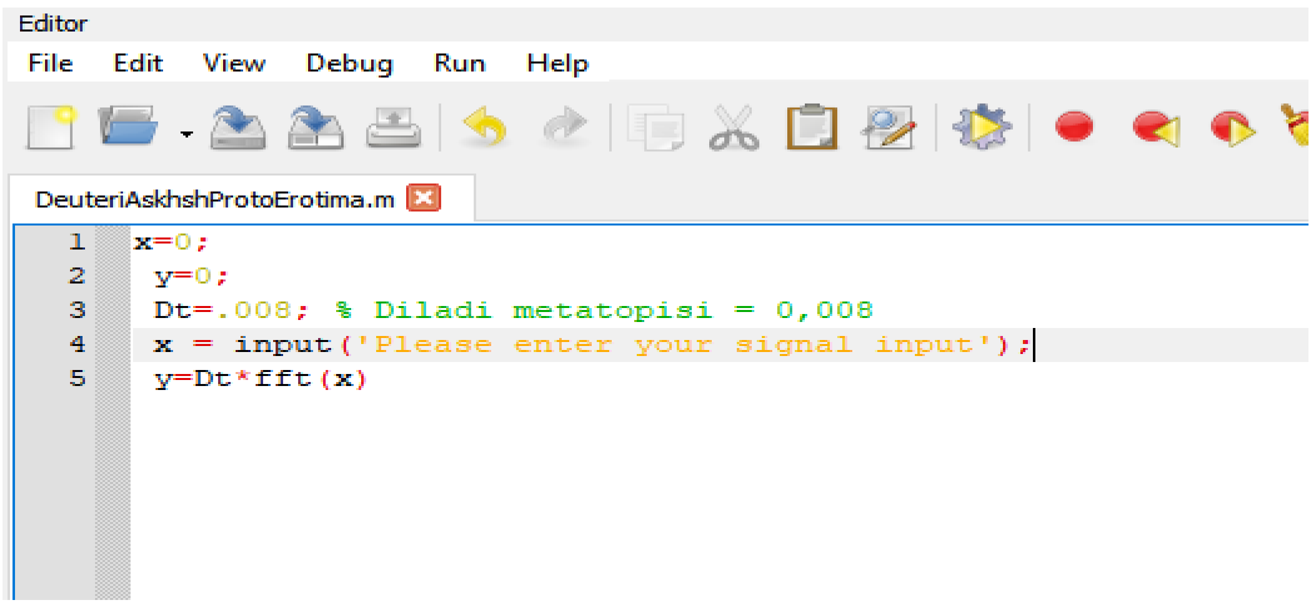
Γ2

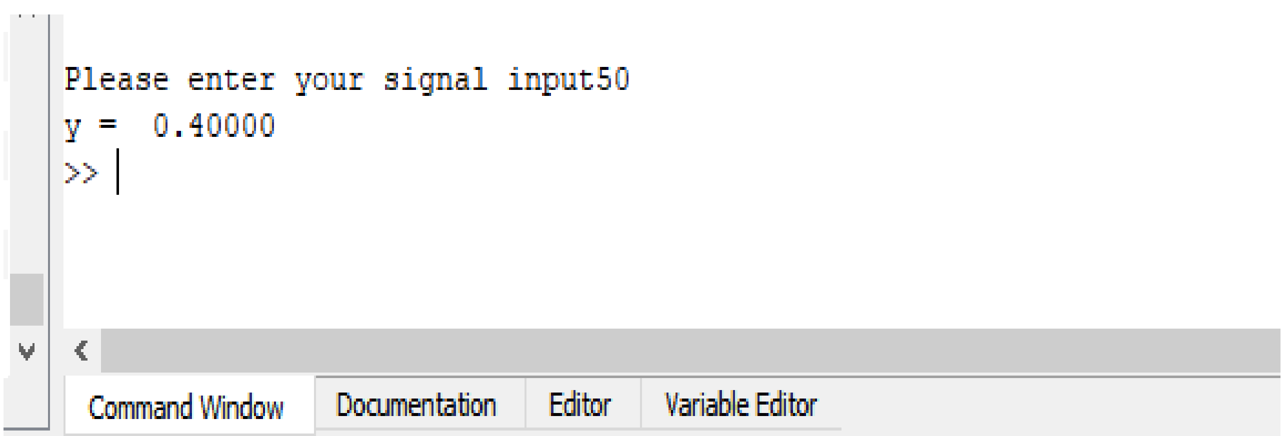
1.

Ο διακριτός μετασχηματισμός Fourier έχει τύπο:

$$x(m) = \sum_{n=1}^{n} x(n) \cdot exp\left(-j \cdot 2 \cdot \pi \cdot (m-1) \cdot (n-1)/N\right)$$

(όπου 1<=m<=N) με N να είναι το μήκος του διανύσματος X





2.

Ο αντίστροφος διακριτός μετασχηματισμός Fourier έχει τύπο:

$$x(n) = (1/N)\sum_{k=1}^{N} X(k)exp(-j \cdot 2 \cdot \pi \cdot (k-1) \cdot (n-1)/N)$$

(όπου 1<=n<=N) με N να είναι το μήκος του διανύσματος X

*DeuteriAskhshDeuteroErotima.m

```
y=0;
y=input('input the Vector ');
n=0;
n=input('input the fundamental frequency ');
Dt=0;
Dt=input('input the factors ');
z=0;
z=(1/Dt).*ifft(y,n)
```

Command Window

```
input the Vector60
input the fundamental frequency 20
input the factors0.05
z =

    60
    60
    60
    60
    60
    60
    60
    60
    60
    60
    60
    60
    60
    60
    60
    60
    60
    60
    60
    60

>> 
```

3.

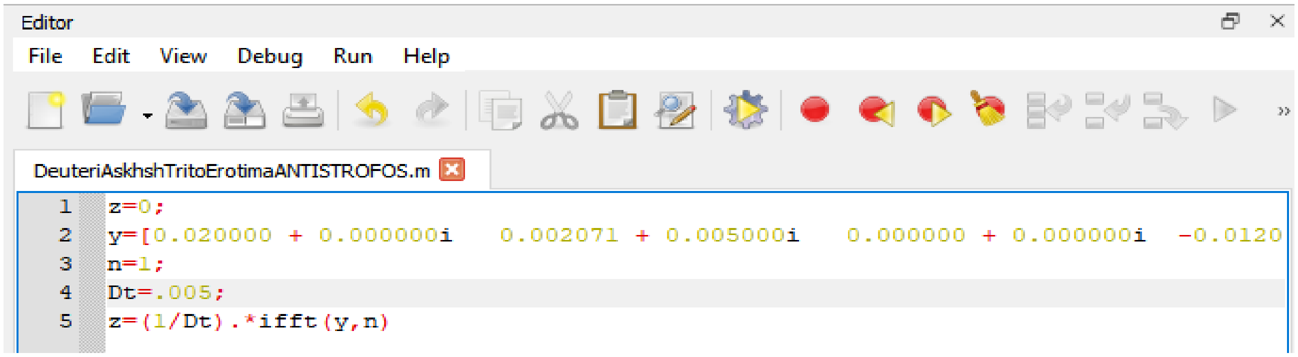
Έχουμε επιλέξει αγγλικό αλφάβητο και το επώνυμό μου ξεκινάει από K(0100 1011).

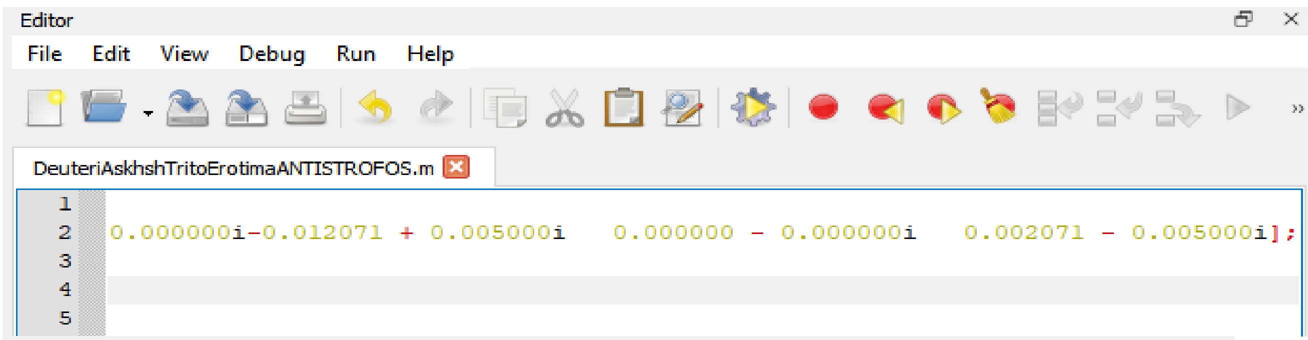
Για τον μετασχηματισμό Fourier έχουμε :

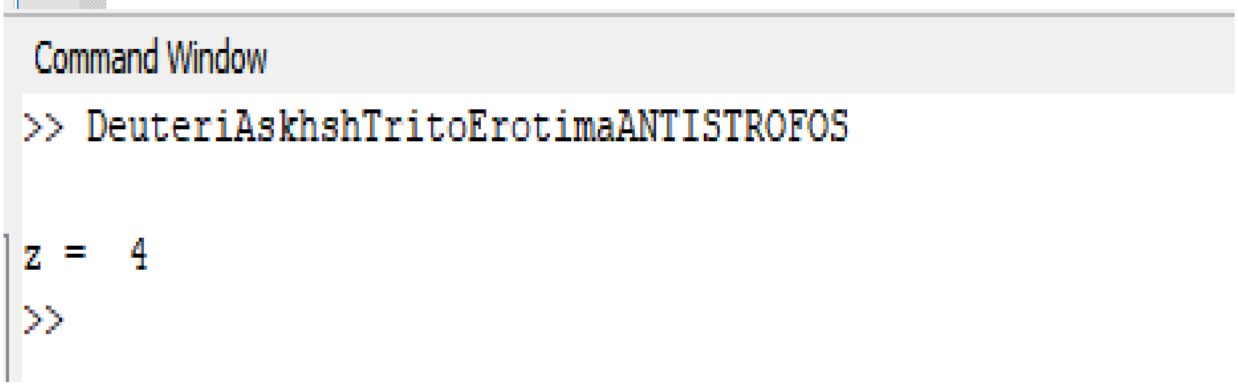
```
Editor
File   Edit   View   Debug   Run   Help
DeuteriAskhshTritoErotimaPROTO.m

x=0;
y=0;
Dt=.005;
x=[0 1 0 0 1 0 1 1];
y=Dt*fft(x)
```

```
>> DeuteriAskhshTritoErotimaPROTO

y =

 Columns 1 through 3:

   0.020000 + 0.000000i   0.002071 + 0.005000i   0.000000 + 0.000000i

 Columns 4 through 6:

  -0.012071 - 0.005000i   0.000000 + 0.000000i  -0.012071 + 0.005000i

 Columns 7 and 8:

   0.000000 - 0.000000i   0.002071 - 0.005000i

>> 
```

Για τον αντίστροφο μετασχηματισμό Fourier έχουμε :







Γ3

1.

Για τις νοτες:

Το 1/8 της νότας έχει διάρκεια 0,25 sec

το ¼=0,5 sec και

το ½ =1 sec.

Η συχνότητα δειγματοληψίας είναι στα 8000 δείγματα ανά δευτερόλεπτο. Άρα θα χρησιμοποιηθεί η **Δt=(1/8)*1024=1/8192 sec(Μετατόπιση)** και η συνάρτηση **f(t)=sin2πνt η οποία θα μηδενίζεται για ν=0 Hz.**

TritiAskhshProtoErotima.m TritiAskhshDeuteroErotima.m

```
t1 = 0:1/8192:0.15;
 a = sin(0*2*pi*t1);
 t2 = 0.15:1/8192:0.25;
 b = sin(220*2^(2/12)*2*pi*t2);
 t3 = 0.25:1/8192:0.5;
 c = sin(0*2*pi*t3);
 t4 = 0.5:1/8192:1.01;
 d = sin(220*2^(6/12)*2*pi*t4);
 t5 = 1.01:1/8192:1.02;
 e = sin(0*2*pi*t5);
 t6 = 1.02:1/8192:2.02;
 f = sin(220*2^(3/12)*2*pi*t6);
 t7 = 2.02:1/8192:2.25;
 g = sin(0*2*pi*t7);
 t8 = 2.25:1/8192:2.53;
 h = sin(220*2^(8/12)*2*pi*t8);
 t9 = 2.53:1/8192:2.54;
 i = sin(0*2*pi*t9);
 t10 = 2.54:1/8192:3.04;
 j = sin(220*2^(8/12)*2*pi*t10);
 t11 = 3.04:1/8192:3.05;
 k = sin(0*2*pi*t11);
 t12 = 3.05:1/8192:4.05;
 l = sin(220*2^(5/12)*2*pi*t12);
 q = [a b c d e f g h i j k l];
 soundsc(q)
```

2

Για ολίσθηση προς τα άνω μιας οχτάδας θα διπλασιάσουμε την διάρκεια της νότας.

File Edit View Debug Run Help

TritiAskhshDeuteroErotimaAno.m TritiAskhshDeuteroErotimaKato.m

```
t1 = 0:1/8192:0.15;
 a = sin(0*2*pi*t1);
 t2 = 0.15:1/8192:0.25;
 b = sin(2*220*2^(2/12)*2*pi*t2);
 t3 = 0.25:1/8192:0.5;
 c = sin(0*2*pi*t3);
 t4 = 0.5:1/8192:1.01;
 d = sin(2*220*2^(6/12)*2*pi*t4);
 t5 = 1.01:1/8192:1.02;
 e = sin(0*2*pi*t5);
 t6 = 1.02:1/8192:2.02;
 f = sin(2*220*2^(3/12)*2*pi*t6);
 t7 = 2.02:1/8192:2.25;
 g = sin(0*2*pi*t7);
 t8 = 2.25:1/8192:2.53;
 h = sin(2*220*2^(8/12)*2*pi*t8);
 t9 = 2.53:1/8192:2.54;
 i = sin(0*2*pi*t9);
 t10 = 2.54:1/8192:3.04;
 j = sin(2*220*2^(8/12)*2*pi*t10);
 t11 = 3.04:1/8192:3.05;
 k = sin(0*2*pi*t11);
 t12 = 3.05:1/8192:4.05;
 l = sin(2*220*2^(5/12)*2*pi*t12);
 q = [a b c d e f g h i j k l];
 soundsc(q)
```

Για ολίσθηση προς τα κάτω μιας οχτάδας θα υποδιπλασιάσουμε την διάρκεια της νότας.

File Edit View Debug Run Help

TritiAskhshDeuteroErotimaAno.m | TritiAskhshDeuteroErotimaKato.m

```
t1 = 0:1/8192:0.15;
 a = sin(0*2*pi*t1);
 t2 = 0.15:1/8192:0.25;
 b = sin(1/2*220*2^(2/12)*2*pi*t2);
 t3 = 0.25:1/8192:0.5;
 c = sin(0*2*pi*t3);
 t4 = 0.5:1/8192:1.01;
 d = sin(1/2*220*2^(6/12)*2*pi*t4);
 t5 = 1.01:1/8192:1.02;
 e = sin(0*2*pi*t5);
 t6 = 1.02:1/8192:2.02;
 f = sin(1/2*220*2^(3/12)*2*pi*t6);
 t7 = 2.02:1/8192:2.25;
 g = sin(0*2*pi*t7);
 t8 = 2.25:1/8192:2.53;
 h = sin(1/2*220*2^(8/12)*2*pi*t8);
 t9 = 2.53:1/8192:2.54;
 i = sin(0*2*pi*t9);
 t10 = 2.54:1/8192:3.04;
 j = sin(1/2*220*2^(8/12)*2*pi*t10);
 t11 = 3.04:1/8192:3.05;
 k = sin(0*2*pi*t11);
 t12 = 3.05:1/8192:4.05;
 l = sin(1/2*220*2^(5/12)*2*pi*t12);
 q = [a b c d e f g h i j k l];
 soundsc(q)
```

3

Για να μετατραπεί η ένταση σε μειούμενη με τον χρόνο πολλαπλασιάσαμε την συνάντηση με την εκθετική συνάρτηση.

File Edit View Debug Run Help

TritiAskhshProtoErotima.m | TritiAskhshTritoErotima.m

```
t1 = 0:1/8192:0.15;
 a = (exp(-1./t1)).*sin(0*2*pi*t1);
 t2 = 0.15:1/8192:0.25;
 b = (exp(-1./t2)).*sin(1/2*220*2^(2/12)*2*pi*t2);
 t3 = 0.25:1/8192:0.5;
 c = (exp(-1./t3)).*sin(0*2*pi*t3);
 t4 = 0.5:1/8192:1.01;
 d = (exp(-1./t4)).*sin(1/2*220*2^(6/12)*2*pi*t4);
 t5 = 1.01:1/8192:1.02;
 e = (exp(-1./t5)).*sin(0*2*pi*t5);
 t6 = 1.02:1/8192:2.02;
 f = (exp(-1./t6)).*sin(1/2*220*2^(3/12)*2*pi*t6);
 t7 = 2.02:1/8192:2.25;
 g = (exp(-1./t7)).*sin(0*2*pi*t7);
 t8 = 2.25:1/8192:2.53;
 h = (exp(-1./t8)).*sin(1/2*220*2^(8/12)*2*pi*t8);
 t9 = 2.53:1/8192:2.54;
 i = (exp(-1./t9)).*sin(0*2*pi*t9);
 t10 = 2.54:1/8192:3.04;
 j = (exp(-1./t10)).*sin(1/2*220*2^(8/12)*2*pi*t10);
 t11 = 3.04:1/8192:3.05;
 k = (exp(-1./t11)).*sin(0*2*pi*t11);
 t12 = 3.05:1/8192:4.05;
 l = (exp(-1./t12)).*sin(1/2*220*2^(5/12)*2*pi*t12);
 q = [a b c d e f g h i j k l];
 soundsc(q)
```

Γ4

Χρησιμοποιήσαμε το image package για τον discrete cosine transformation.

TetatrtiAskhsh.m

```
img = imread('mochart.jpg');
image(img);
imshow(img);
img=double(img)/255;
 img=rgb2gray(img);
 subplot(211)
 imshow(img)
 title('Normal iamge');
 img_dct=dct(img);
 img_pow=(img_dct).^2;
img_pow=img_pow(:);
[B,index]=sort(img_pow);
B=flipud(index);
index=flipud(index);
compressed_dct=zeros(size(img));
coeff = 500;
for k=1:coeff
  compressed_dct(index(k))=img_dct(index(k));
endfor
img_dct=idct2(compressed_dct);
subplot(212)
imshow(img_dct)
title('DCT compress photo')
imwrite(img_dct,'compress.bmp')
```

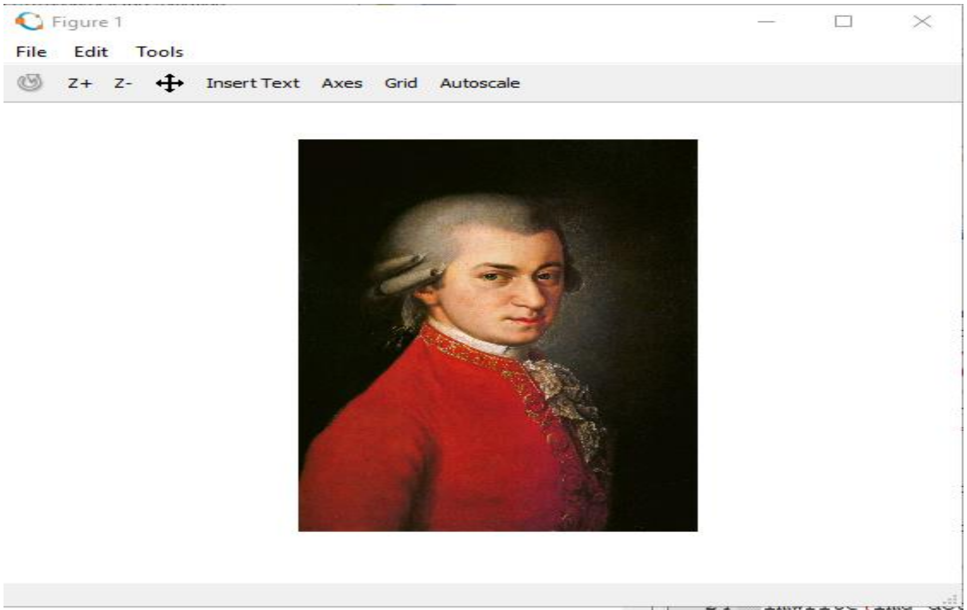


Figure 1
File Edit Tools
Z+ Z- Insert Text Axes Grid Autoscale

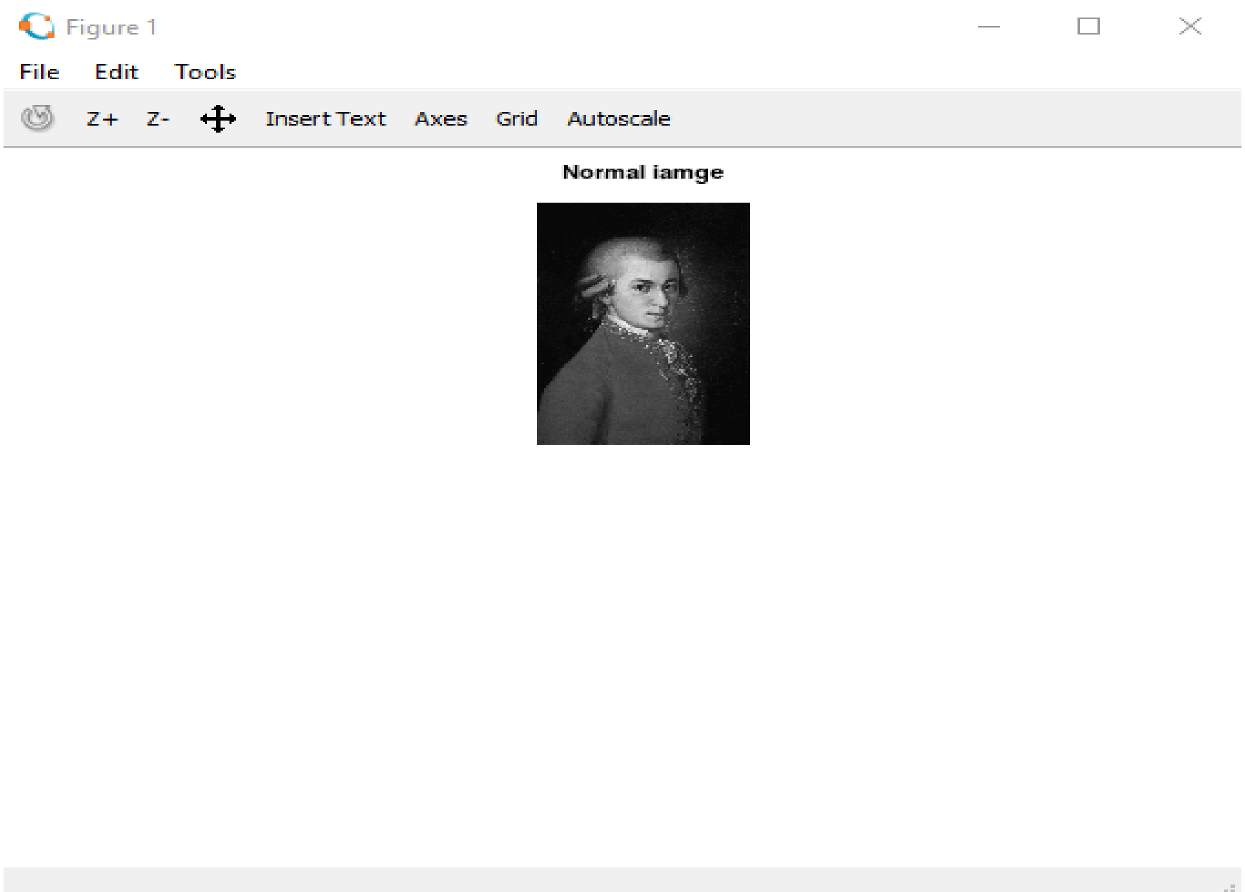
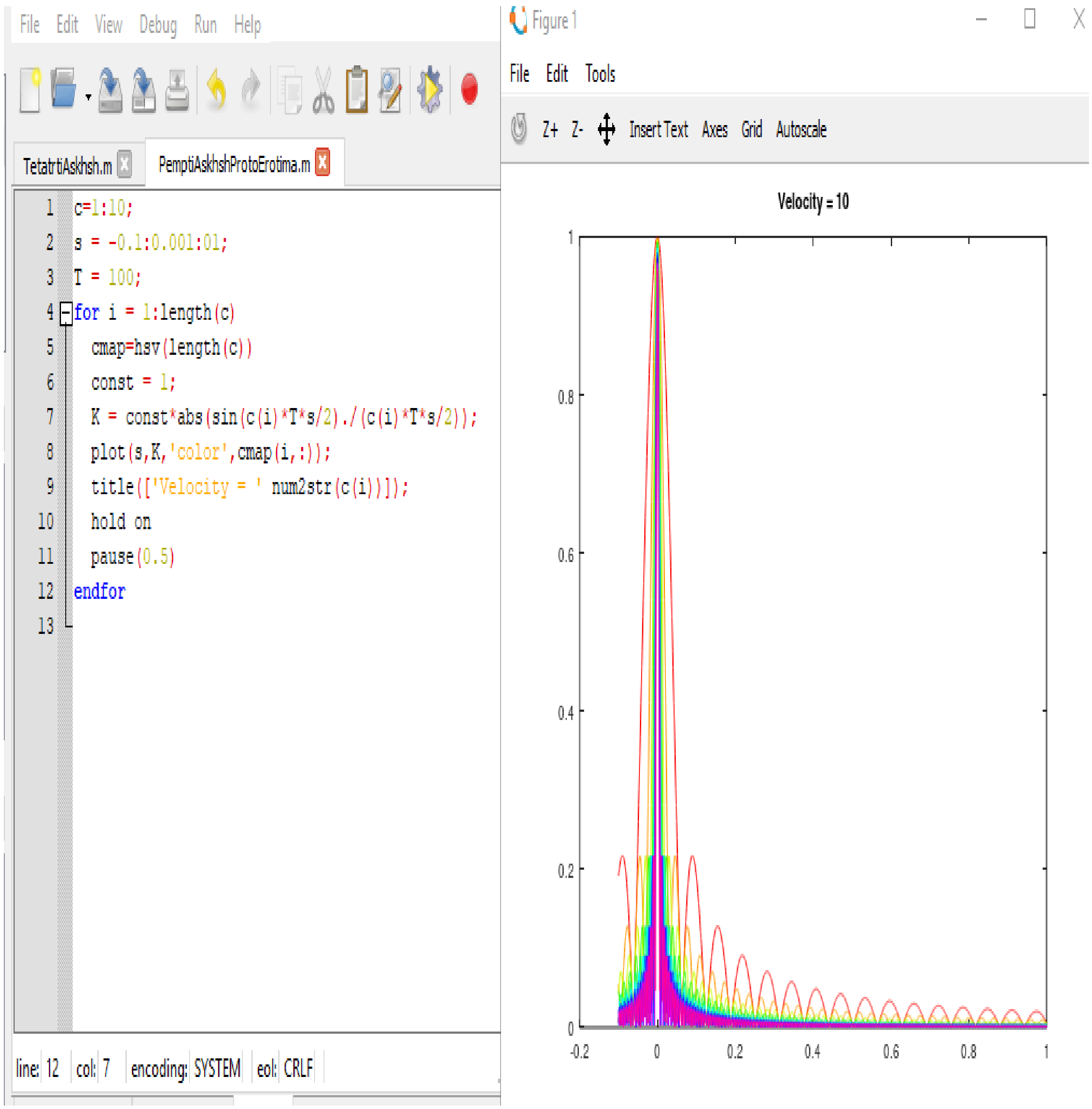


Figure 1
File Edit Tools
Z+ Z- Insert Text Axes Grid Autoscale
Normal iamge

Γ5

1) Αποθηκεύουμε την τιμή του μέτρου της συνάρτησης στην μεταβλητή K.

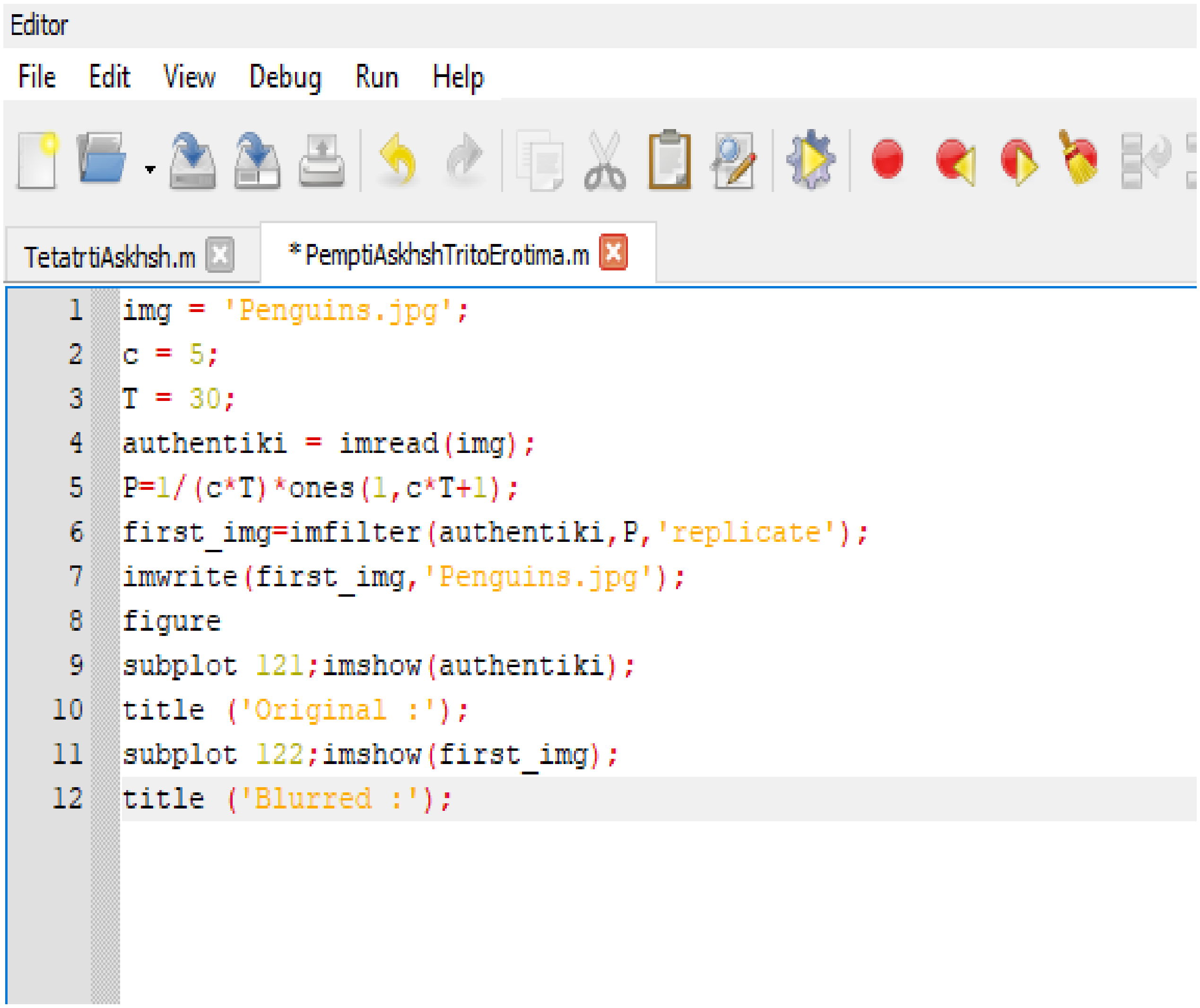
Έτσι, θα εμφανίζονται τα κύματα τιμών για κάθε διαφορετική τιμή του μέτρου c από το 1 έως το 10.



2)

Η αύξηση της γραμμικής ταχύτητας c μικραίνει τις τιμές του μέτρου της συνάρτησης μεταφοράς K. Αυτό δικαιολογείται από το γεγονός πως κατά την διάρκεια εξέτασης της θέσης του πρώτου μηδενισμού στην πρώτη λύση της εξίσωσης ως προς το μέτρο s , ο μηδενισμός παρατηρείται συγκριτικά λιγότερες φορές από όταν το s δέχεται μεγαλύτερες τιμές .Επομένως , καταλήγουμε στο συμπέρασμα ότι το φαινόμενο του θολώματος γίνεται πιο έντονο και η εικόνα πιο δυσδιάκριτη.

3)




```
img = 'Penguins.jpg';
c = 5;
T = 30;
authentiki = imread(img);
P=1/(c*T)*ones(1,c*T+1);
first_img=imfilter(authentiki,P,'replicate');
imwrite(first_img,'Penguins.jpg');
figure
subplot 121;imshow(authentiki);
title ('Original :');
subplot 122;imshow(first_img);
title ('Blurred :');
```

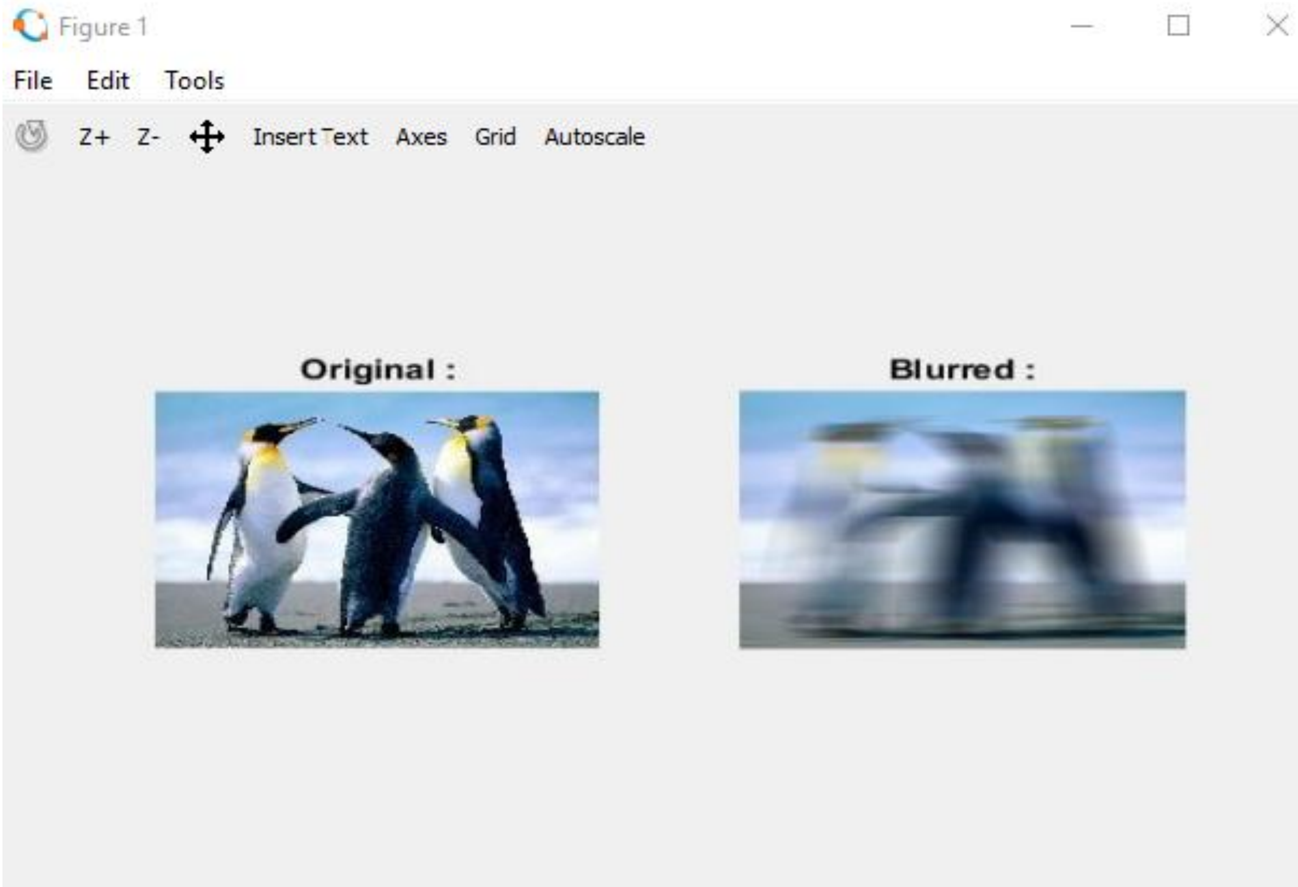


Figure 1
File Edit Tools
Z+ Z- Insert Text Axes Grid Autoscale
Original :
Blurred :