

Εργασία Σήματα και Συστήματα 2021

ΟΝΟΜΑΤΕΠΩΝΥΜΟ	ΑΡΙΘΜΟΣ ΜΗΤΡΩΟΥ
Βιτάκης Αθανάσιος	Π19247
Αυγερινός Χρήστος	Π19020
Παναγιωτόπουλος Δημήτριος	Π19130

Περιεχόμενα

Εργασία Σήματα και Συστήματα 2021.....	1
Γ1	2
Ερώτημα Γ-1.1:	2
Ερώτημα Γ-1.2	3
Ερώτημα Γ-1.3	4
Ερώτημα Γ-1.4	5
Ερώτημα Γ-1.5	6
Ερώτημα Γ-1.6	7
Γ2	8
Ερώτημα Γ-2.1	8
Ερώτημα Γ-2.2	9
Ερώτημα Γ-2.3	10
Γ3	11
Ερώτημα Γ-3.1	11
Ερώτημα Γ-3.2	12
Ερώτημα Γ-3.3	12
Γ4	13
Γ5	14
Ερώτημα Γ-5.1	14
Ερώτημα Γ-5.2:	15
Ερώτημα Γ-5.3:	16

Γ1

Ερώτημα Γ-1.1:

Σήμα συνεχούς χρόνου: $x(t) = \cos(100\pi t) + \cos(200\pi t) + \sin(500\pi t)$

Γνωρίζουμε ότι $x(t) = 2\pi S t$

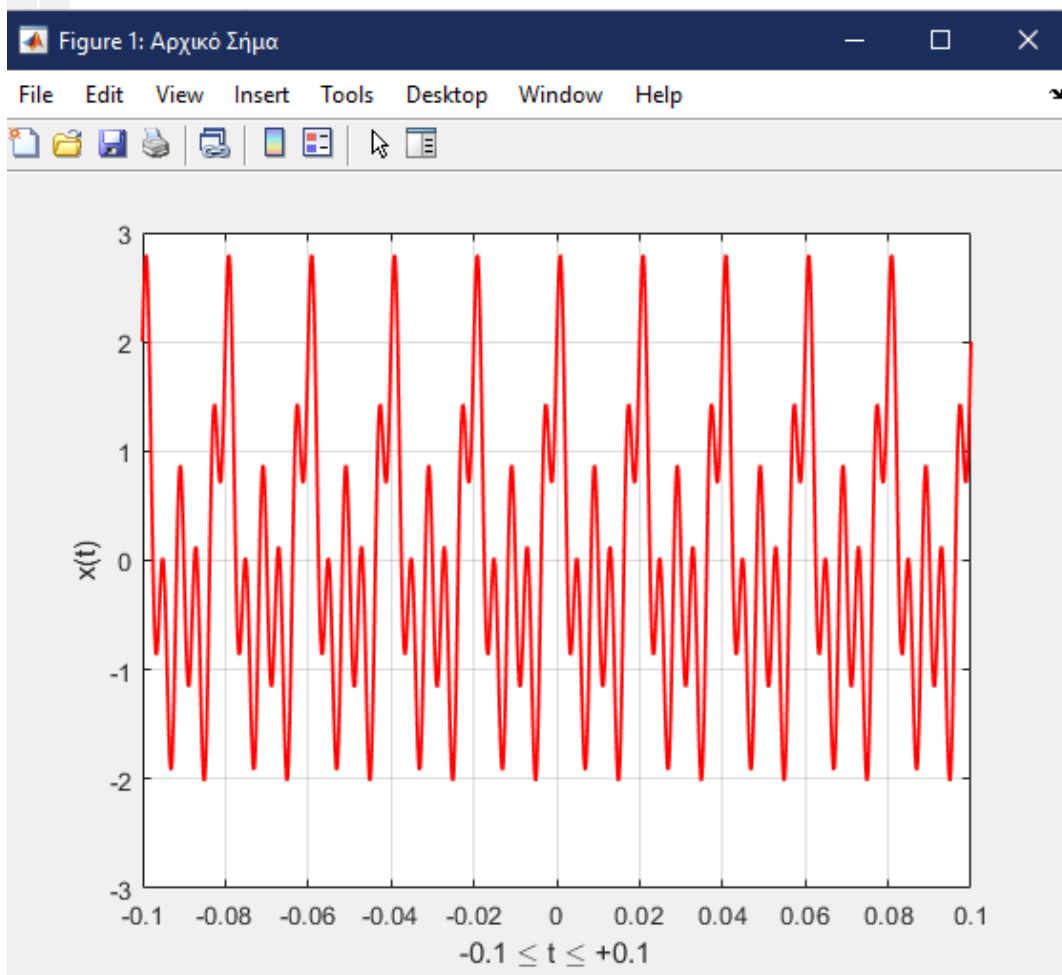
Άρα διαιρώντας με το 2 βρίσκουμε τις εξής συχνότητες:

Στο δοσμένο σήμα έχουμε $S_1=50$, $S_2=100$ και $S_3=250$

Άρα σύμφωνα με το θεώρημα δειγματοληψίας ο χρόνος κατά τον οποίον πρέπει να πάρουμε δείγματα είναι $1/(2 \cdot S) = 1/500$ (ή $T_S=0.002$ sec)

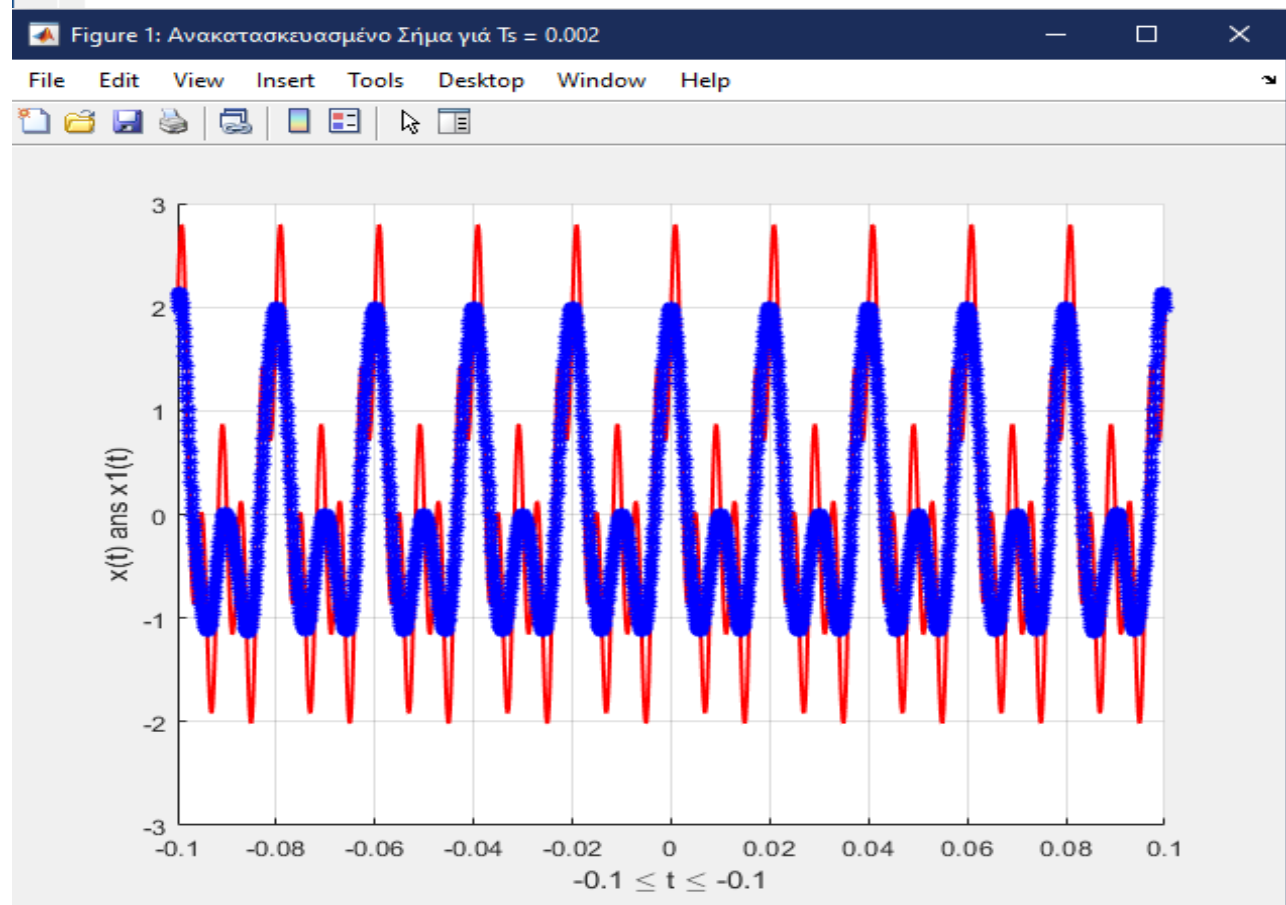
Ερώτημα Γ-1.2

```
1
2   % Original Signal Representation.
3 -   dt = 0.0001;
4 -   t_max = 0.1;
5 -   t = -t_max:dt:t_max;
6 -   x = cos(100*pi*t) + cos(200*pi*t) + sin(500*pi*t);
7 -   figure('Name','Αρχικό Σήμα');
8 -   plot(t,x,'-r','LineWidth',1.3);
9 -   xlabel('-0.1 \leq t \leq +0.1');
10 -  ylabel('x(t)');
11 -  grid on
12 |
```



Ερώτημα Γ-1.3

```
1 - Ts = 0.002;
2 - Nmax = t_max / Ts;
3 - nl = -Nmax:1:Nmax;
4 - xs = cos(100*pi*nl*Ts) + cos(200*pi*nl*Ts) + sin(500*pi*nl*Ts);
5 - x1 = zeros(1,length(t));
6 - for k = 1:1:length(t)
7 -     x1(k) = xs * sinc((t(k)-nl*Ts)/Ts)';
8 - end
9 - figure('Name','Ανακατασκευασμένο Σήμα για Ts = 0.002');
10 - hold on
11 - plot(t,x,'-r','LineWidth',1.5);
12 - plot(t,x1,'*b','LineWidth',1.2);
13 - xlabel('-0.1 \leq t \leq 0.1');
14 - ylabel('x(t) ans x1(t)');
15 - grid on
16
17 |
```

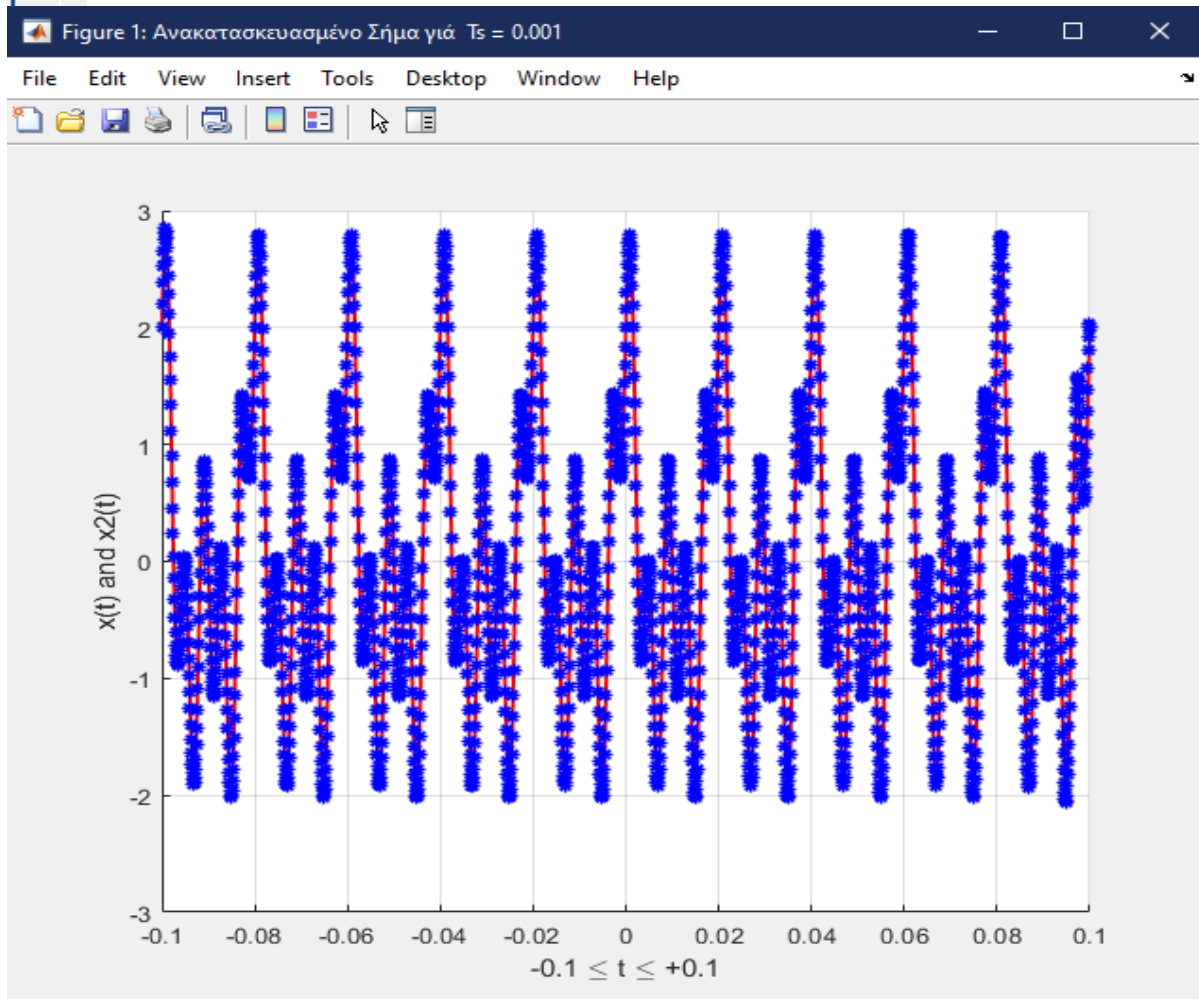


Ερώτημα Γ-1.4

```

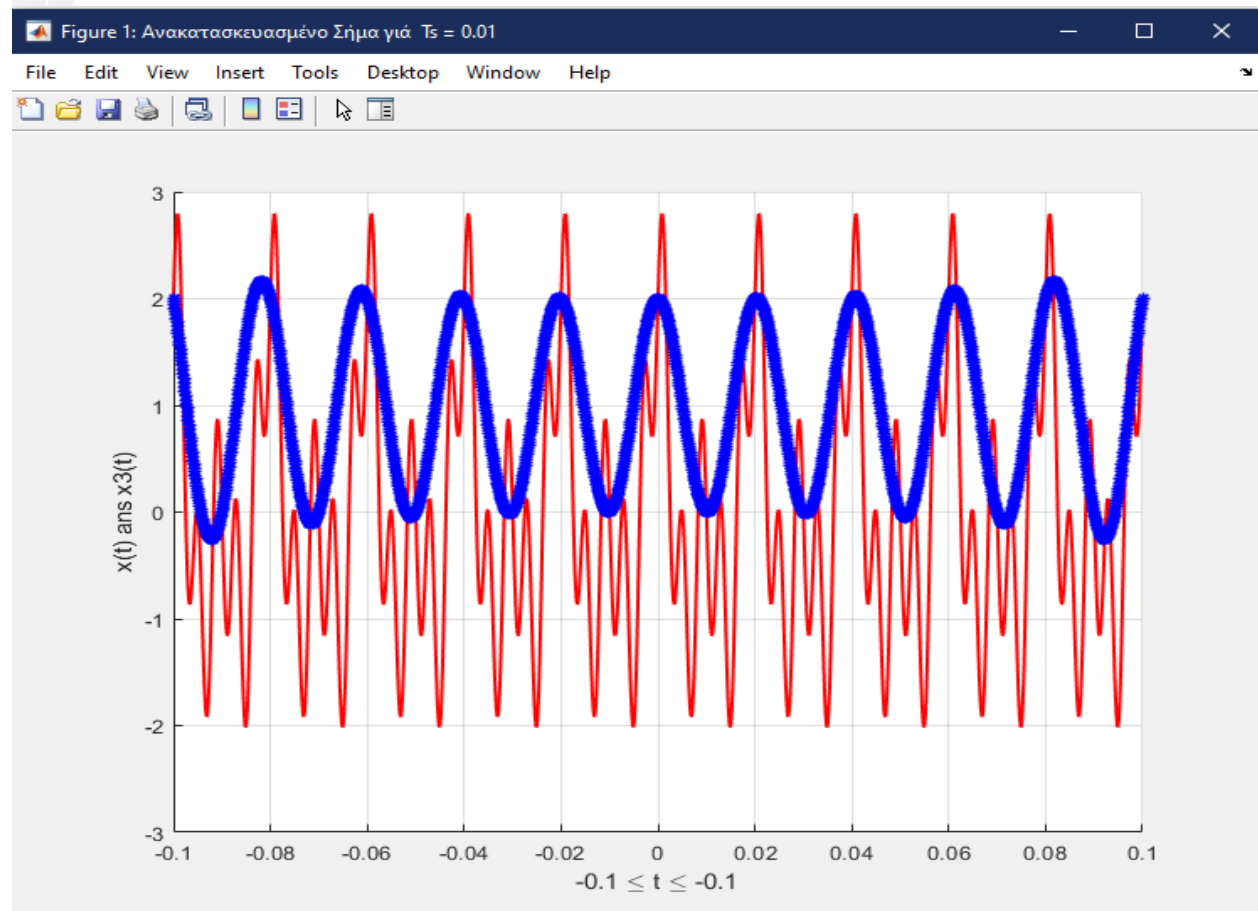
1 - Ts = 0.001;
2 - Nmax = t_max / Ts;
3 - n2 = -Nmax:1:Nmax;
4 - xs = cos(100*pi*n2*Ts) + cos(200*pi*n2*Ts) + sin(500*pi*n2*Ts);
5 - x2 = zeros(1,length(t));
6
7 - for k = 1:length(t)
8 -     x2(k) = xs * sinc((t(k)-n2*Ts)/Ts);
9 - end
10
11 - figure('Name','Ανακατασκευασμένο Σήμα για Ts = 0.001');
12 - hold on
13 - plot(t,x,'-r','LineWidth',1.5);
14 - plot(t,x2,'*b','LineWidth',1.2);
15 - xlabel('-0.1 \leq t \leq +0.1');
16 - ylabel('x(t) and x2(t)');
17 - grid on
18

```



Ερώτημα Γ-1.5

```
1 - Ts = 0.01;
2 - Nmax = t_max / Ts;
3 - n3 = -Nmax:1:Nmax;
4 - xs = cos(100*pi*n3*Ts) + cos(200*pi*n3*Ts) + sin(500*pi*n3*Ts);
5 - x3 = zeros(1,length(t));
6
7 - for k = 1:1:length(t)
8 -     x3(k) = xs * sinc((t(k)-n3*Ts)/Ts)';
9 - end
10
11 - figure('Name','Ανακατασκευασμένο Σήμα για Ts = 0.01');
12 - hold on
13 - plot(t,x,'-r','LineWidth',1.5);
14 - plot(t,x3,'*b','LineWidth',1.2);
15 - xlabel('-0.1 \leq t \leq 0.1');
16 - ylabel('x(t) ans x3(t)');
17 - grid on
```



Ερώτημα Γ-1.6

Παρατηρούμε ότι το ανακατασκευασμένο σήμα για $T_s = 0.002$ δεν μας δίνει πίσω το αρχικό διότι έχει εισαχθεί θόρυβος.

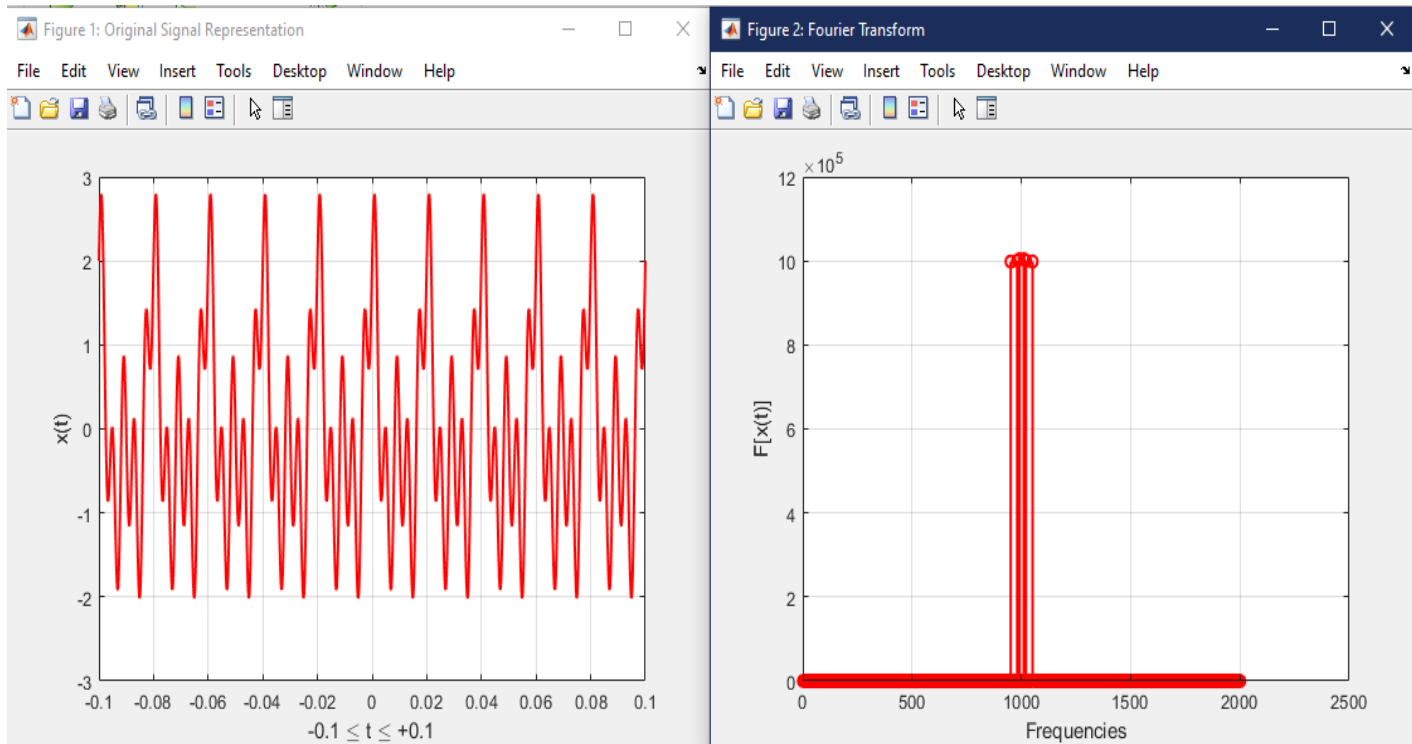
Έπειτα αφότου πήραμε μεγαλύτερη συχνότητα δειγματοληψίας ($T_s=0.001$) καταφέραμε και ανακατασκευάσαμε πλήρως το αρχικό σήμα.

Τέλος παίρνοντας συχνότητα μικρότερη από αυτή που υπολογίσαμε από το ερώτημα 1 παρατηρούμε ότι το σήμα που ανακατασκευάσαμε είναι εντελώς διαφορετικό με το αρχικό.

Γ2

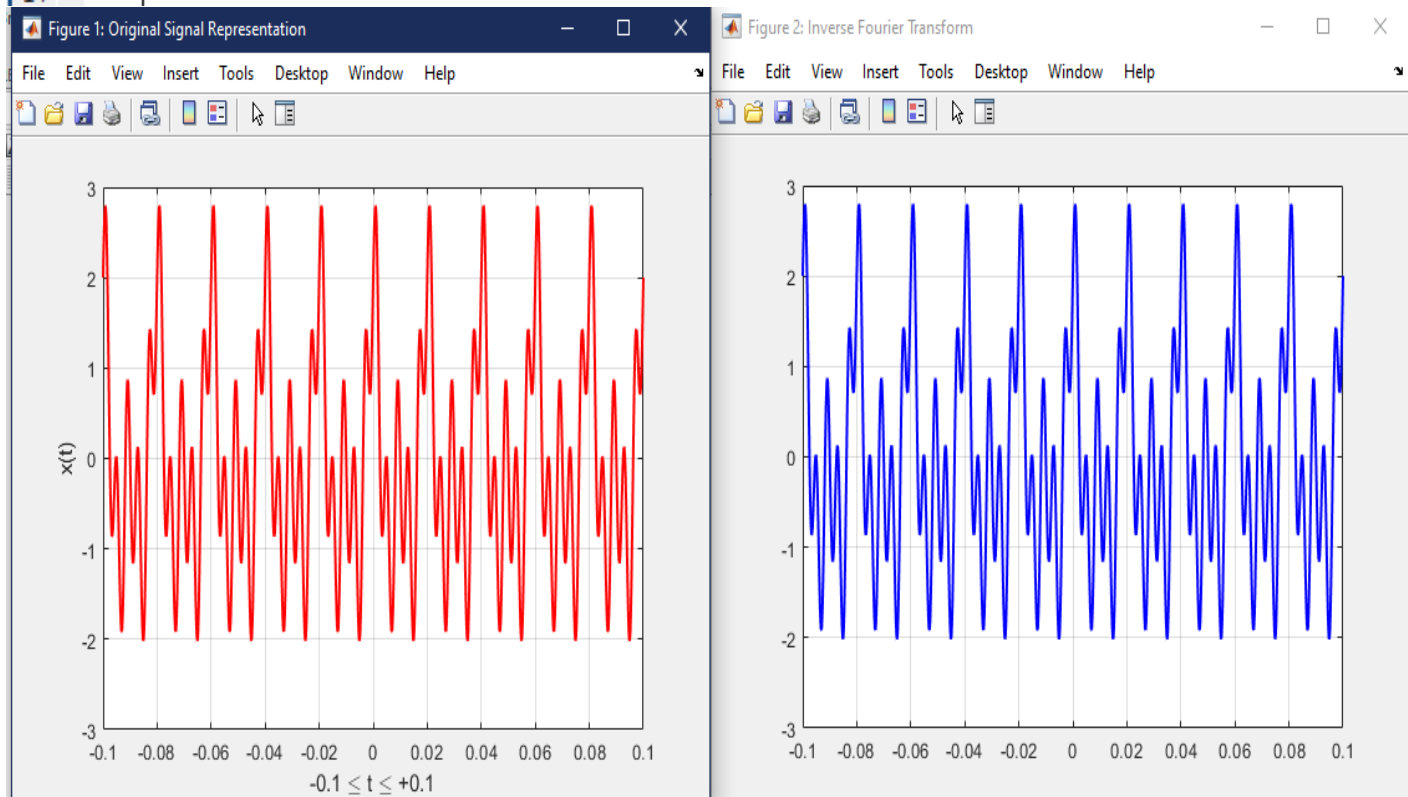
Ερώτημα Γ-2.1

```
1 % Original Signal Representation.
2 dt = 0.0001;
3 t_max = 0.1;
4 t = -t_max:dt:t_max;
5 x = cos(100*pi*t) + cos(200*pi*t) + sin(500*pi*t);
6 figure('Name','Original Signal Representation');
7 plot(t,x,'-r','LineWidth',1.5);
8 xlabel('-0.1 \leq t \leq +0.1');
9 ylabel('x(t)');
10 grid on
11
12 % FFT
13 Fx = fft(x);
14 Wx = Fx .* conj(Fx);
15 Wx = fftshift(Wx);
16 figure('Name','Fourier Transform');
17 stem(Wx,'-r','LineWidth',1.5);
18 xlabel('Frequencies');
19 ylabel('F[x(t)]');
20 grid on
21 |
```



Ερώτημα Γ-2.2

```
1 % Original Signal Representation.
2 dt = 0.0001;
3 t_max = 0.1;
4 t = -t_max:dt:t_max;
5 x = cos(100*pi*t) + cos(200*pi*t) + sin(500*pi*t);
6 figure('Name','Original Signal Representation');
7 plot(t,x,'-r','LineWidth',1.5);
8 xlabel('-0.1 \leq t \leq +0.1');
9 ylabel('x(t)');
10 grid on
11
12 %IFFT
13 IFx = ifft(Fx);
14 figure('Name','Inverse Fourier Transform');
15 plot(t,IFx,'-b','LineWidth',1.5);
16 grid on
17
```

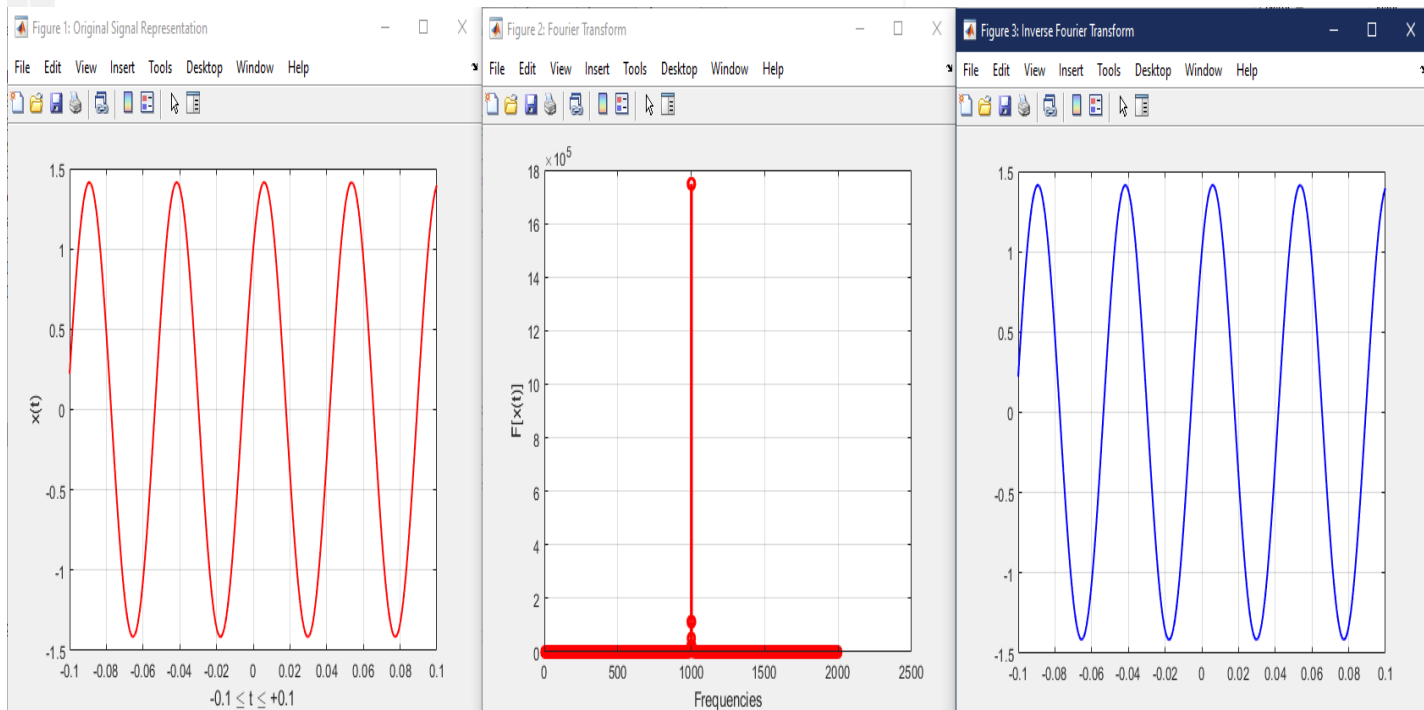


Ερώτημα Γ-2.3

```

1 % Original Signal Representation.
2 dt = 0.0001;
3 t_max = 0.1;
4 t = -t_max:dt:t_max;
5 %B to Ascii (Hex) = 42
6 x = cos(42*pi*t) + sin(42*pi*t);
7 figure('Name','Original Signal Representation');
8 plot(t,x,'-r','LineWidth',1.5);
9 xlabel('-0.1 \leq t \leq +0.1');
0 ylabel('x(t)');
1 grid on
2
3 % FFT
4 Fx = fft(x);
5 Wx = Fx .* conj(Fx);
6 Wx = fftshift(Wx);
7 figure('Name','Fourier Transform');
8 stem(Wx,'-r','LineWidth',1.5);
9 xlabel('Frequencies');
0 ylabel('F[x(t)]');
1 grid on
2
3 %IFFT
4 IFx = ifft(Fx);
5 figure('Name','Inverse Fourier Transform');
6 plot(t,IFx,'-b','LineWidth',1.5);
7 grid on

```



Γ3

```

1  function note = g3function(selected_note, duration, fs,volume)
2
3  -   t = 0:(1/fs):duration;
4
5  -       switch selected_note
6  -           case 'A ', f = 220;
7  -           case 'A#', f = 220*2^(1/12);
8  -           case 'B ', f = 220*2^(2/12);
9  -           case 'C ', f = 220*2^(3/12);
10 -          case 'C#', f = 220*2^(4/12);
11 -          case 'D ', f = 220*2^(5/12);
12 -          case 'D#', f = 220*2^(6/12);
13 -          case 'E ', f = 220*2^(7/12);
14 -          case 'F ', f = 220*2^(8/12);
15 -          case 'F#', f = 220*2^(9/12);
16 -          case 'G ', f = 220*2^(10/12);
17 -          case 'G#', f = 220*2^(11/12);
18 -       end
19
20 -   note = volume*sin(2*pi*f*t);
21
22 - end

```

Ερώτημα Γ-3.1

```

1
2  -   Notes_Array = ['A ','A#','B ','C ','C#','D ','D#','E ','F ','F#','G ','G#'];
3   %Happy Birthday Song
4  -   Song = ['C ','C ','D ','C ','F ','E '
5  -           'C ','C ','D ','C ','G ','F '
6  -           'C ','C ','C ','A ','F ','E ','D '
7  -           'A#','A#','A ','F ','G ','F ','F '];
8
9  -   fs = 8000;
10 -   Dt = 0.4;
11 -   for i = 1:length(Song)
12 -       sound(g3function(Song(i,1:end),Dt,fs,0.1));
13 -       sound(0);
14 -       pause(0.3)
15 -   end
16

```

Ερώτημα Γ-3.2

```

1
2 - Notes_Array = ['A ','A#','B ','C ','C#','D ','D#','E ','F ','F#','G ','G#'];
3 %Happy Birthday Song
4 - Song = ['C ','C ','D ','C ','F ','E '
5           'C ','C ','D ','C ','G ','F '
6           'C ','C ','C ','A ','F ','E ','D '
7           'A#','A#','A ','F ','G ','F ','F '];
8 - fs = 8000;
9 - Dt = 0.1;
10 - for i = 1:length(Song)
11 -     sound(g3function(Song(i,1:end),i*Dt,fs,0.1));
12 -     sound(0);
13 -     pause(0.3)
14 - end
15

```

Ερώτημα Γ-3.3

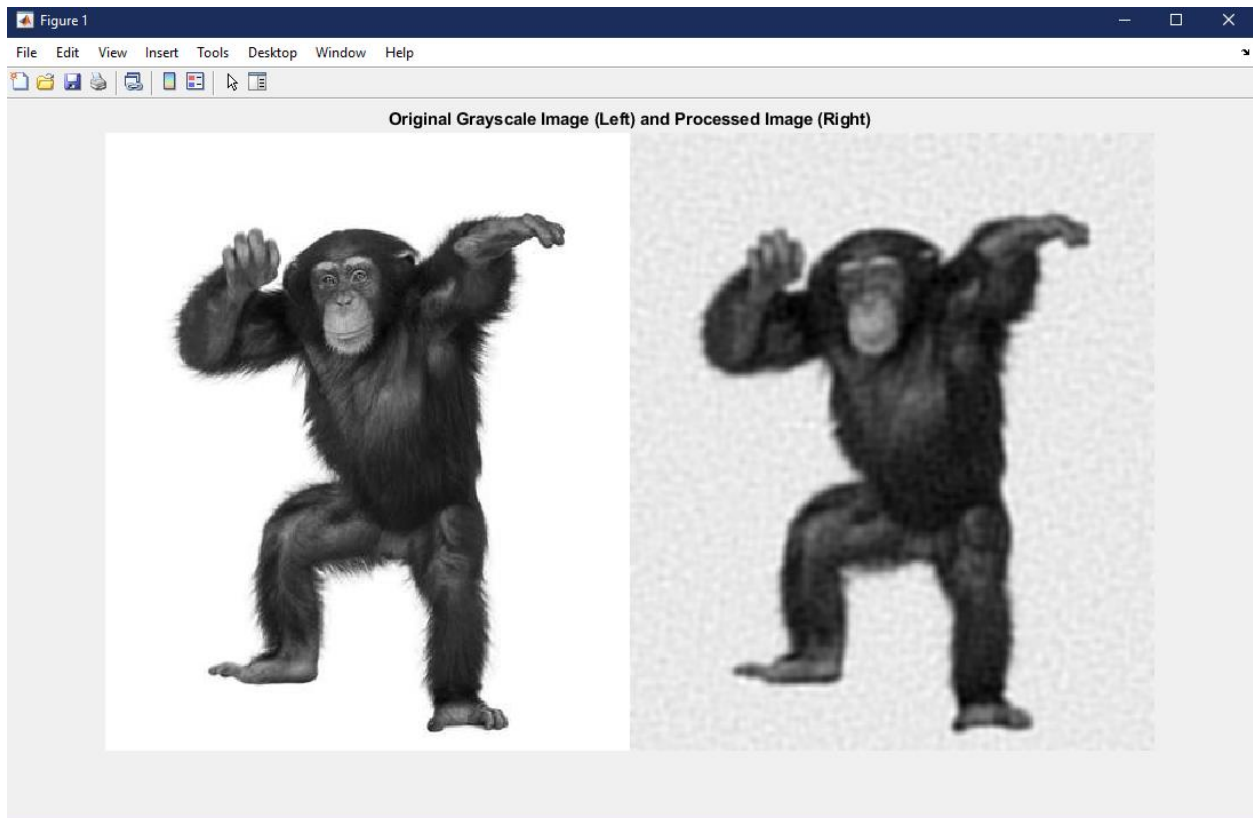
```

2 - Notes_Array = ['A ','A#','B ','C ','C#','D ','D#','E ','F ','F#','G ','G#'];
3 %Happy Birthday Song
4 - Song = ['C ','C ','D ','C ','F ','E '
5           'C ','C ','D ','C ','G ','F '
6           'C ','C ','C ','A ','F ','E ','D '
7           'A#','A#','A ','F ','G ','F ','F '];
8 - fs = 8000;
9 - Nt = 0.2;
10 - for i = 1:length(Song)
11 -     sound(g3function(Song(i,1:end),Nt,fs,(13-i)/10));
12 -     sound(0);
13 -     pause(0.3)
14 - end

```

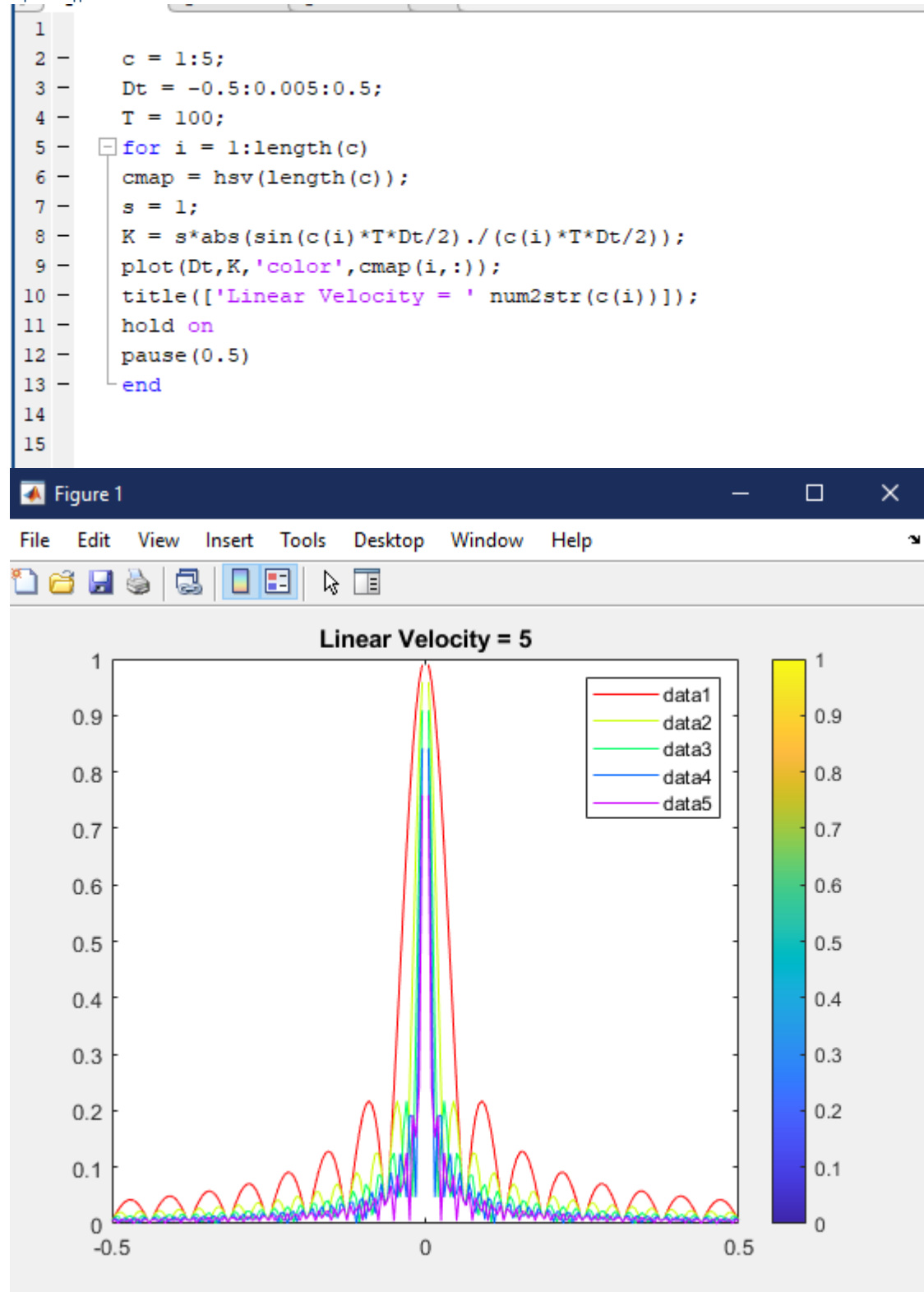
4

```
1 - RGB = imread('monkey.jpg');  
2 - I = im2gray(RGB);  
3 - J = dct2(I);  
4 - J(abs(J) < 50) = 0; %compression by 50%  
5 - K = idct2(J);  
6 - K = rescale(K);  
7 - montage({I,K})  
8 - title('Original Grayscale Image (Left) and Processed Image (Right)');
```



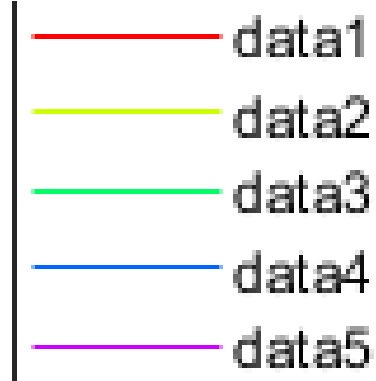
Γ5

Ερώτημα Γ-5.1



Ερώτημα Γ-5.2:

Παρατηρούμε ότι όσο πιο γρήγορο το c τόσο πιο γρήγορα μηδενίζεται η συνάρτηση μεταφοράς



Ερώτημα Γ-5.3:

```
1 - image = 'monkey.jpg';
2 - output = 'monkeyblurred.jpg';
3
4 - c = 1;
5 - T = 50; % 50% blurred
6
7 - Input = imread(image);
8
9 - K = 1/(c*T)*ones(1,c*T+1);
10
11 - Blurred_image = imfilter(Input,K,'replicate');
12
13 - imwrite(Blurred_image, output);
14 - montage({Input,Blurred_image})
15 - title('Original Image (Left) and Blurred Image (Right)');
```

