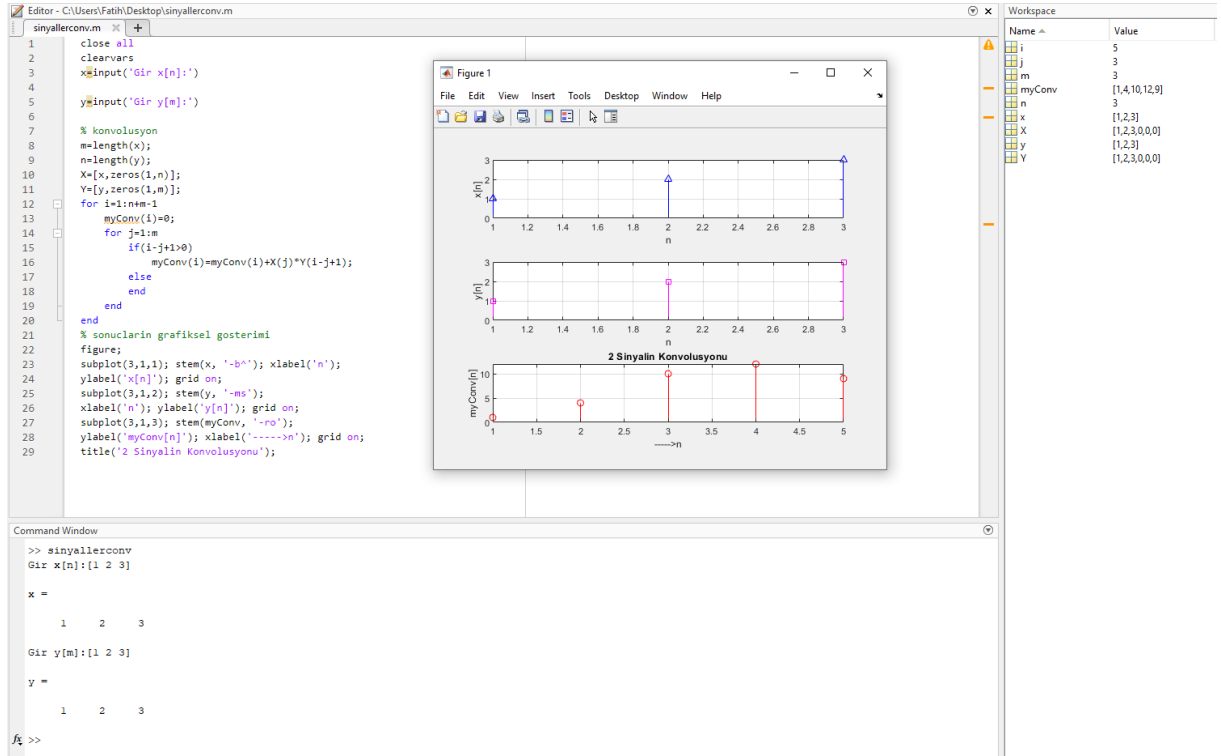
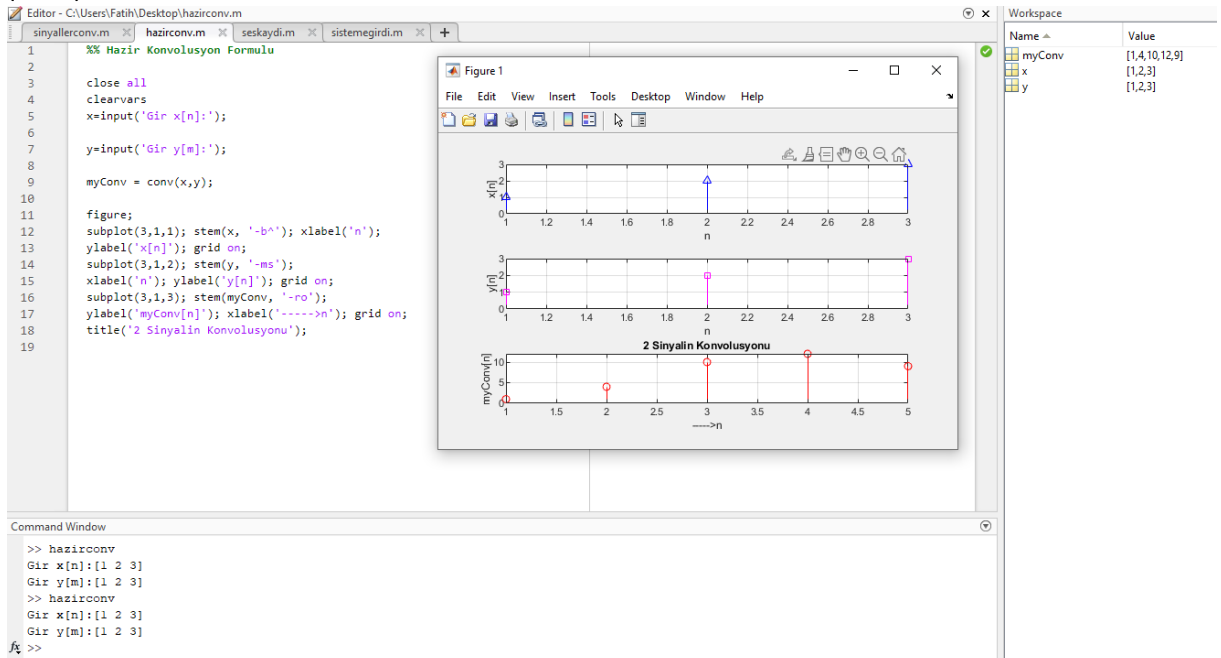


- 1- Konvolüsyon toplamı algoritması yazarak 2 ayırık zamanlı işaretin konvolüsyon toplamlarını buluyoruz. (yazılan algoritma)



- 2- Bu işaretleri ve alınan sonuçları grafiksel ve vektörel olarak gösteriyoruz. (hazır)



- 3- Kendi sesimizi X1 ve X2 değişkenine sırasıyla 5 saniye ve 10 saniye olarak verilen kod örneği ile kaydediyoruz.

```

1 %% 5 Saniye Ses Kaydetme
2 recObj = audiorecorder; %% kayıt başlatma nesnesi
3 disp('Start speaking.') %% ekrana mesaj
4 recordblocking(recObj, 5); %% kayıt işlemi
5 disp('End of Recording. '); %% ekrana mesaj
6 x1 = getaudiodata(recObj); %% kaydedilen sesi x1 değişkeninde saklama
7
8 %% 10 Saniye Ses Kaydetme
9 recObj = audiorecorder; %% kayıt başlatma nesnesi
10 disp('Start speaking.') %% ekrana mesaj
11 recordblocking(recObj, 10); %% kayıt işlemi
12 disp('End of Recording. '); %% ekrana mesaj
13 x = getaudiodata(recObj); %% kaydedilen sesi x2 değişkeninde saklama

```

Command Window

```

>> hazirconv
Gir x[n]:[1 2 3]
Gir y[m]:[1 2 3]
>> hazirconv
Gir x[n]:[1 2 3]
Gir y[m]:[1 2 3]
>> hazirconv
Gir x[n]:[1 2 3]
Gir y[m]:[1 2 3]
>> seskaydi
Start speaking.
End of Recording.
Start speaking.
End of Recording.
f1 >>

```

Workspace

| Name | Value |
|--------|-------------------|
| myConv | [1,4,10,12,9] |
| recObj | 1x1 audiorecorder |
| x | 80000x1 double |
| x1 | 40000x1 double |
| y | [1,2,3] |

- 4- Bu ses dosyalarını verilen sisteme giriş olarak vererek kendi yazdığımız konvolüsyon toplamı algoritması ve dilde bulunan hazır konvolüsyon toplamı fonksiyonu olarak 2 farklı şekilde sonuçları kaydediyoruz.

```

15 %% Sisteme Girdi Olarak Verme
16
17 n=X1;
18 y(n) = x(n) + 0.4 * x(n-400) + 0.4 * x(n-800);
19 y(n) = deneme5;
20 Y1 = conv(X1,deneme5);
21
22 n=X2;
23 y(n) = x(n) + 0.4 * x(n-400) + 0.4 * x(n-800);
24 y(n) = deneme10;
25 Y2 = conv(X2,deneme10);
26
27 x=X1;
28 y=deneme5;
29 m=length(x);
30 n=length(y);
31 X=[x,zeros(1,n)];
32 Y=[y,zeros(1,m)];
33 for i=1:n+m-1
34     myConv(i)=0;
35     for j=1:m
36         if(i-j+1>0)
37             myConv(i)=myConv(i)+X(j)*Y(i-j+1);
38         else
39             end
40     end
41 end
42
43 My_Y1 = myConv;
44
45 x=X2;
46 y=deneme10;
47 m=length(x);
48 n=length(y);
49 X=[x,zeros(1,n)];
50 Y=[y,zeros(1,m)];
51 for i=1:n+m-1
52     myConv(i)=0;
53     for j=1:m
54         if(i-j+1>0)
55             myConv(i)=myConv(i)+X(j)*Y(i-j+1);
56         else
57             end
58     end
59 end
60
61 My_Y2 = myConv;
62

```

- 5- 'sound' komutu ile ilk girişleri ve elde ettiğimiz sonuçlar arasındaki farklılıkları yorumluyoruz.

```
Editor - C:\Users\Fatih\Desktop\seslendirme.m
sinyallerconv.m x hazirconv.m x seskaydi.m x sistemegirdi.m x seslendirme.m x +
1 %%Giriş Verilerini Seslendirme
2
3 sound(X1)
4 sound(X2)
5 sound(Y1)
6 sound(Y2)
7 sound(My_Y1)|
8 sound(My_Y2)
```

SON HALİ

```
Editor - C:\Users\Fatih\Desktop\20011610.m
sinyallerconv.m x hazirconv.m x seskaydi.m x sistemegirdi.m x seslendirme.m x 20011610.m x +
1 %% Yazarak Konvolusyon Kullanma
2
3 close all
4 clearvars
5 x=input('Gir x[n]:')
6
7 y=input('Gir y[m]:')
8
9 % konvolusyon
10 m=length(y);
11 n=length(x);
12 X=[x,zeros(1,n)];
13 Y=[y,zeros(1,m)];
14 for i=1:n+m-1
15     myConv(i)=0;
16     for j=1:m
17         if(i-j+1>0)
18             myConv(i)=myConv(i)+X(j)*Y(i-j+1);
19         else
20             end
21     end
22 end
23 % sonucun grafiksel gosterimi
24 figure;
25 subplot(3,1,1); stem(x, '-b^'); xlabel('n');
26 ylabel('x[n]'); grid on;
27 subplot(3,1,2); stem(y, '-ms');
28 xlabel('n'); ylabel('y[n]'); grid on;
29 subplot(3,1,3); stem(myConv, '-ro');
30 ylabel('myConv[n]'); xlabel('----->n'); grid on;
31 title('2 Sinyalin Konvolusyonu');
32
33 %% Hazir Konvolusyon Formulu
34
35 close all
36 clearvars
37 x=input('Gir x[n]:');
38
39 y=input('Gir y[m]:');
40
41 myConv = conv(x,y);
42
43 msg = ['Result of conv is'];
44 disp(msg)
45 disp(myConv)
46
47 figure;
48 subplot(3,1,1); stem(x, '-b^'); xlabel('n');
49 ylabel('x[n]'); grid on;
50 subplot(3,1,2); stem(y, '-ms');
51 xlabel('n'); ylabel('y[n]'); grid on;
52 subplot(3,1,3); stem(myConv, '-ro');
53 ylabel('myConv[n]'); xlabel('----->n'); grid on;
54 title('2 Sinyalin Konvolusyonu');
55
56 %% 5 Saniye Ses Kaydetme
57 recObj = audiorecorder; %% kayıt başlatma nesnesi
58 disp('Start speaking.') %% ekrana mesaj
59 recordblocking(recObj, 5); %% kayıt işlemi
60 disp('End of Recording.') %% ekrana mesaj
61 X1 = getaudiodata(recObj); %% kaydedilen sesi x1 değişkeninde saklama
```

```
Editor - C:\Users\Fatih\Desktop\20011610.m
sinyallerconv.m x1 hazirconv.m x2 seskaydi.m x3 sistemegirdi.m x4 seslendirme.m 20011610.m x5 +
63 %% 1. Sinyalın Ses Kaydetme
64 recObj = audiorecorder; %% kayıt başlatma nesnesi
65 disp('Start speaking.') %% ekrana mesaj
66 recordblocking(recObj, 10); %% kayıt işlemi
67 disp('End of Recording. '); %% ekrana mesaj
68 X2 = getaudiodata(recObj); %% kaydedilen sesi x2 değişkeninde saklama
69
70 %% Sisteme Girdi Olarak Verme
71
72 n=X1;
73 y(n) = x(n) + 0.4 * x(n-400) + 0.4 * x(n-800);
74 y(n) = deneme5;
75 Y1 = conv(X1,deneme5);
76
77 n=X2;
78 y(n) = x(n) + 0.4 * x(n-400) + 0.4 * x(n-800);
79 y(n) = deneme10;
80 Y2 = conv(X2,deneme10);
81
82 x=X1;
83 y=deneme5;
84 m=length(x);
85 n=length(y);
86 X=[x,zeros(1,n)];
87 Y=[y,zeros(1,m)];
88 for i=1:n+m-1
89     myConv(i)=0;
90     for j=1:m
91         if(i-j+1>0)
92             myConv(i)=myConv(i)+X(j)*Y(i-j+1);
93         else
94             end
95     end
96 end
97
98 My_Y1 = myConv;
99
100 x=X2;
101 y=deneme10;
102 m=length(x);
103 n=length(y);
104 X=[x,zeros(1,n)];
105 Y=[y,zeros(1,m)];
106 for i=1:n+m-1
107     myConv(i)=0;
108     for j=1:m
109         if(i-j+1>0)
110             myConv(i)=myConv(i)+X(j)*Y(i-j+1);
111         else
112             end
113     end
114 end
115
116 My_Y2 = myConv;
117
118 %% Giriş Verilerini Seslendirme
119
120
121 sound(X1)
122 sound(X2)
123 sound(Y1)
124 sound(Y2)
125 sound(My_Y1)
```