Name: basil khowaja and yahya faraz

# Task 1

## Correlation between the signals x=[2 2 4 5] and y=[4 5 4 0]

```
x = [2 2 4 5];
y = [4 5 4 0];
[r, k] = xcorr(x, y)
```

```
r = 1×7
    -0.0000    8.0000    18.0000    34.0000    48.0000    41.0000    20.0000

k = 1×7
    -3    -2    -1    0    1    2    3
```

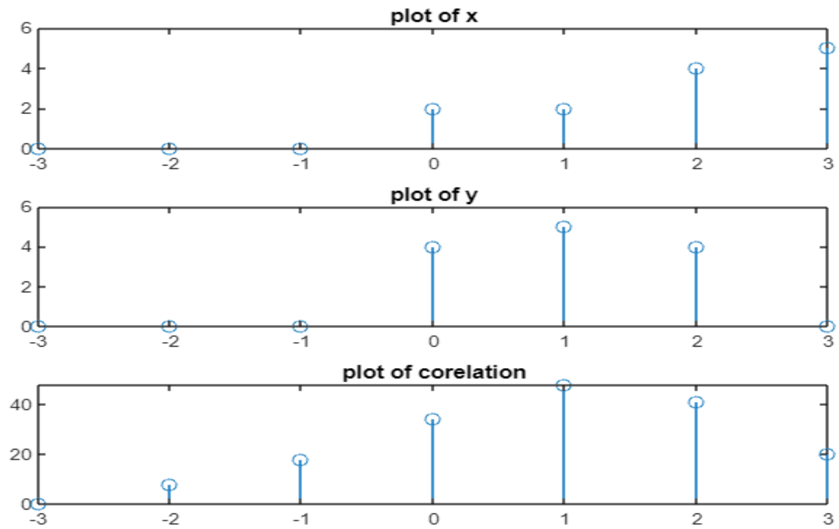
a] The range of lags returned will be from **−N+1** to **N−1**, where N is the length of the input signal. In this case, N=4. So **the range of k (lags) is -3 <= k <= +3**.

b] The length of correlation is the sum of both lengths, which is **n1+n2-1**. In this case, **n1=n2=4** So, the final length is **2n-1 = 2(8)-1 = 7**.

c] **CODE:**

```
x_n = [zeros(1, 3) x];
y_n = [zeros(1, 3) y];
figure;
subplot 311; stem(k, x_n);
title ("plot of x");
subplot 312; stem(k, y_n);
title ("plot of y");
subplot 313; stem(k, r);
title ("plot of corelation");
```

Name: basil khowaja and yahya faraz




d] From observation, correlation is maximum when **lag=1** (autocorrelation is more than 40).

# Task 2

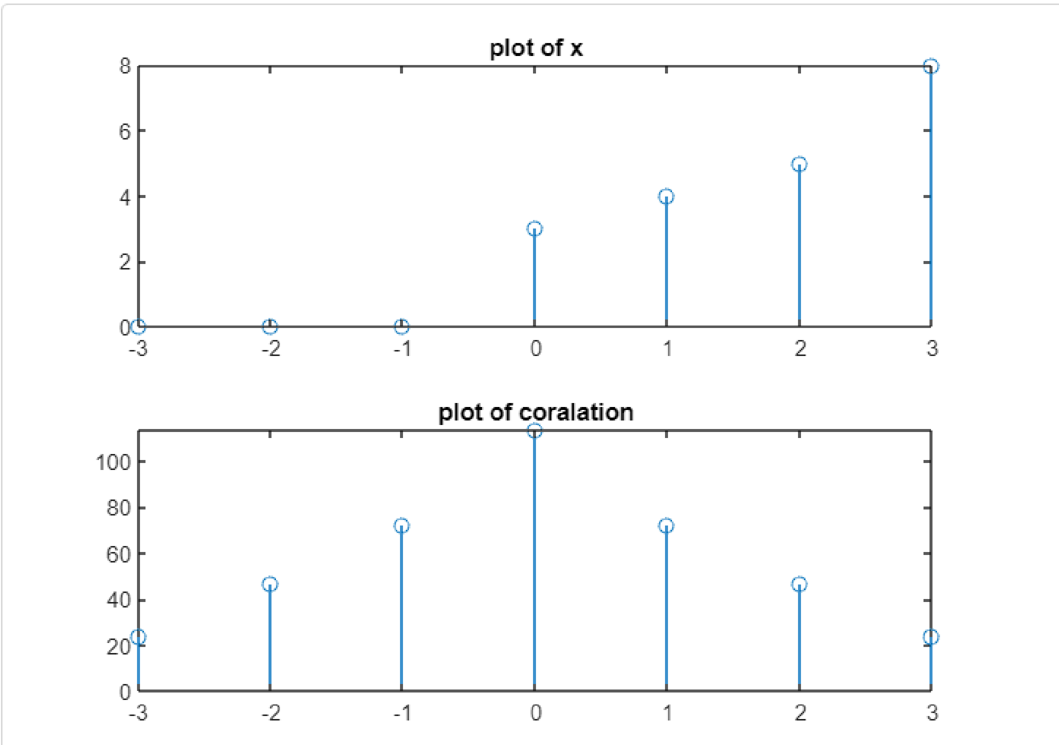
```
x = [3 4 5 8];
[r, k] = xcorr(x, x)
x_n = [zeros(1, 3) x];
figure;
subplot 211; stem(k, x_n);
title ("plot of x");
subplot 212; stem(k, r);
title ("plot of coralation");
```

Plot:

Name: basil khowaja and yahya faraz

```
r = 1×7
     24.0000    47.0000    72.0000   114.0000    72.0000    47.0000    24.0000

k = 1×7
      -3    -2    -1     0     1     2     3
```




The **maximum value** in the autocorrelation plot is **when there is no lag**, showing that the signal is the most similar to itself without a time shift. As the lag varies from 0, the **autocorrelation decreases equally** and is **symmetric about the y-axis**.

Name: basil khowaja and yahya faraz

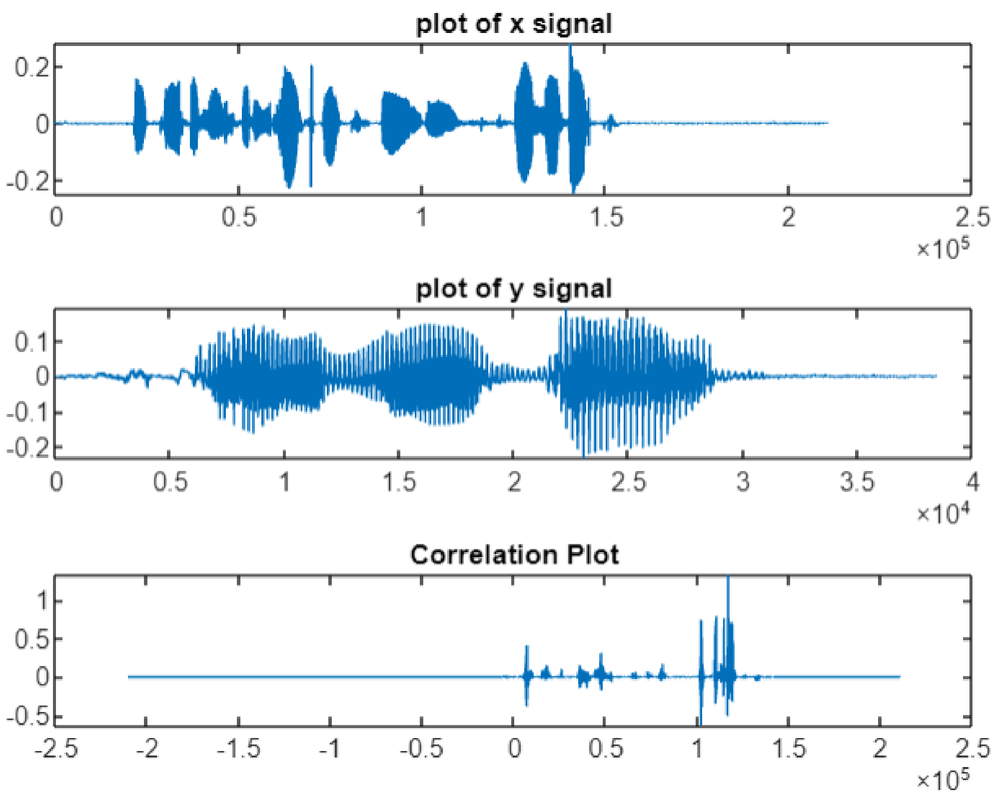
# Task 3

```
[y , Fs_y] = audioread('speech_segment.wav'); % segment of signal we want to
%search
[x , Fs_x] = audioread('speech.wav'); % complete signal or speech
sound(y, Fs_y) % sends the signal in vector y (with sample frequency Fs˙y)
%to the speaker on PC
pause(2) %to insert a pause of 2 seconds between the two audio files
sound(x, Fs_x)

[correlation , lag] = xcorr(x,y);
cmax = max(abs(correlation));
Nonlinear_Effect = (1.1* correlation/cmax).^3 ;
figure;
subplot 311; plot(x);
title ("plot of x signal");
subplot 312; plot(y);
title ("plot of y signal");
subplot 313; plot(lag, Nonlinear_Effect);
title ("Correlation Plot");

[a, index] = max(Nonlinear_Effect)
location=lag(index)
sound(x(location:end),Fs_x);
```

Name: basil khowaja and yahya faraz




```
a = 1.3310
index = 327381
location = 116794
```

## observation:

Because of this normalization the output of the co relation is between -1 and 1, the 1.1 factor is used where there is peak of the correlation. If we apply correlation without the normalization then thea amplitude of the correlation would be very large.

In this task we are extracting the audio segment from the given sample by using the correlation method. The peak of the correlation will tell where the segment is and then we got its index using the max command and then played the sound of the original audio to the end to get that sound sample.

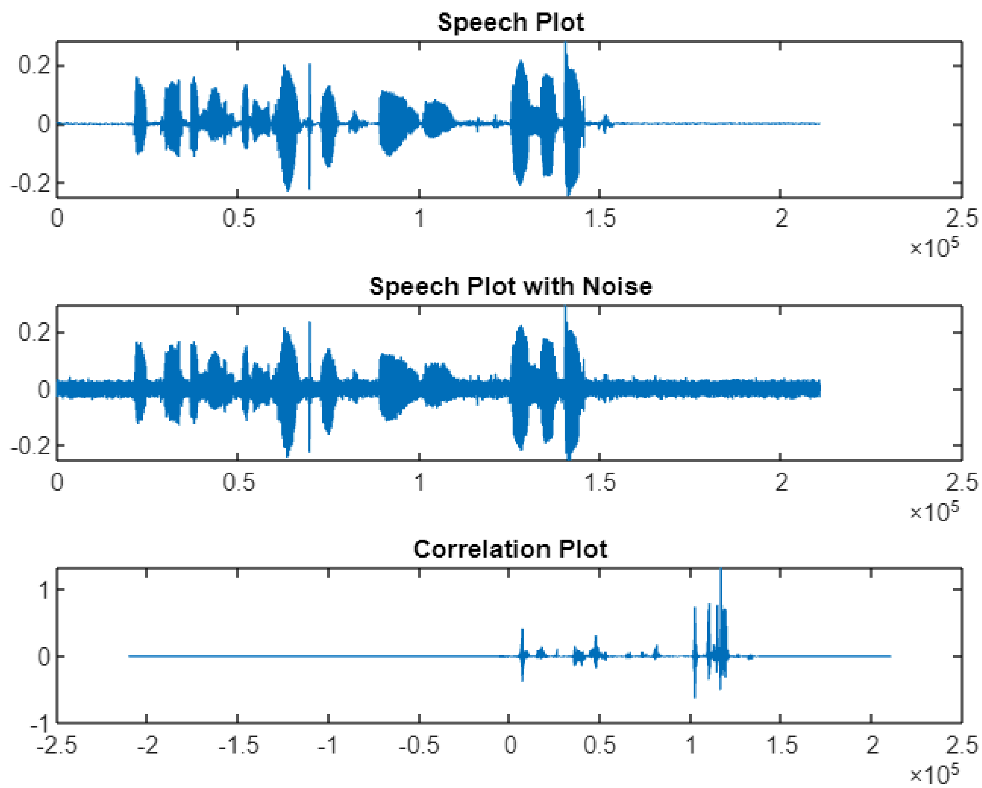
Name: basil khowaja and yahya faraz

# Task 4

```
x_noise = randn(size(x));
x_new = x + x_noise/100;
[correlation , lag] = xcorr(x,y);
cmax = max(abs(correlation));
Nonlinear_Effect = (1.1* correlation/cmax).^3 ;
figure;
subplot 311; plot(x);
title ("Speech Plot");
subplot 312; plot(x_new);
title ("Speech Plot with Noise");
subplot 313; plot(lag, Nonlinear_Effect);
title ("Correlation Plot");
[a, index] = max(Nonlinear_Effect)
location=lag(index)
sound(x(location:end),Fs_x);

x_new_1 = x + x_noise/90;
sound (x_new_1, Fs_x)
Px = (sum(x.^2))/length(x) %original signal
Pn = (sum(x_new_1.^2))/length(x) %signal in which we putted noise and then divided by 100
SNR = Px/Pn
```

Name: basil khowaja and yahya faraz




```
a = 1.3310
index = 327381
location = 116794

Px = 8.5178e-04
Pn = 9.7444e-04
SNR = 0.8741
```

Name: basil khowaja and yahya faraz

Task 5 (postlab):

```
[x, Fs_x] = audioread('main_audio.wav');
[o1, Fs_o1] = audioread('okay_1.wav');
[o2, Fs_o2] = audioread('okay_2.wav');
[o3, Fs_o3] = audioread('okay_4.wav');
[o4, Fs_o4] = audioread('okay_1.wav');
pause(2)
sound(x, Fs_x)
pause(2)
sound(o1, Fs_o1)
pause(2)
sound(o2, Fs_o2)
pause(2)
sound(o3, Fs_o3)
pause(2)
sound(o4, Fs_o4)
[correlation1 , lag1] = xcorr(x,o1);
cmax1 = max(abs(correlation1));
Nonlinear_Effect1 = (1.1* correlation1/cmax1).^3 ;
[correlation2 , lag2] = xcorr(x,o2);
cmax2 = max(abs(correlation2));
Nonlinear_Effect2 = (1.1* correlation2/cmax2).^3 ;
[correlation3 , lag3] = xcorr(x,o3);
cmax3 = max(abs(correlation3));
Nonlinear_Effect3 = (1.1* correlation3/cmax3).^3 ;
[correlation4 , lag4] = xcorr(x,o4);
cmax4 = max(abs(correlation4));
```

Name: basil khowaja and yahya faraz

```
Nonlinear_Effect4 = (1.1* correlation4/cmax4).^3 ;

figure; plot(x);

title ("Main Audio");

figure;

subplot 411; plot(Nonlinear_Effect1);

title ("signal 1 Plot");

subplot 412; plot(Nonlinear_Effect2);

title ("signal 2 Plot");

subplot 413; plot(Nonlinear_Effect3);

title ("signal 3 Plot");

subplot 414; plot(Nonlinear_Effect4);

title ("signal 4 Plot");

threshold = min([cmax1 cmax2 cmax3 cmax4])
```

```
if (cmax1 > threshold)

   disp("Signal 1: phrase detected");

else

   disp("Signal 1: phrase not detected");

end
```

```
if (cmax2 > threshold)

   disp("Signal 2: phrase detected");

else

   disp("Signal 2: phrase not detected");

end
```

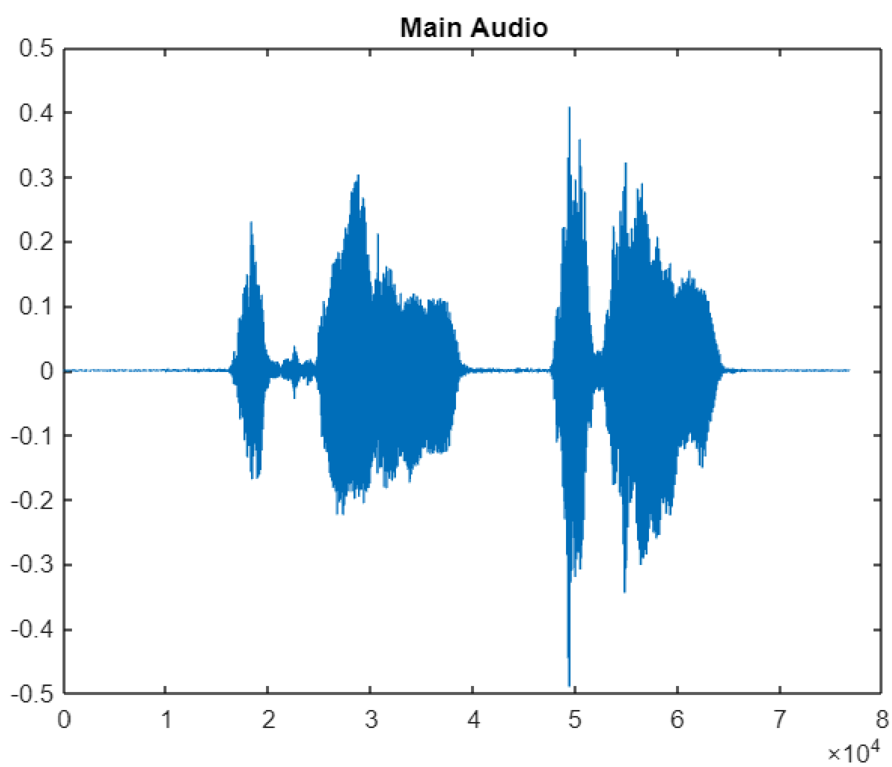
```
if (cmax3 > threshold)

   disp("Signal 3: phrase detected");
```

Name: basil khowaja and yahya faraz

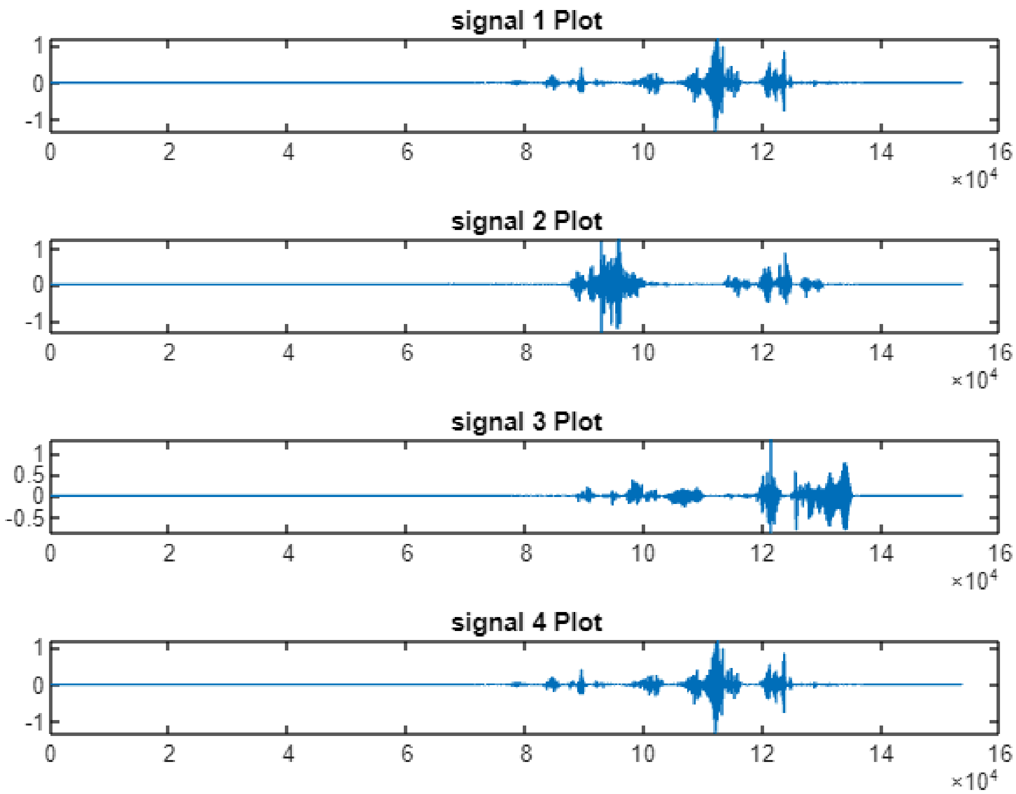
```
else
   disp("Signal 3: phrase not detected");
end
```

```
if (cmax4 > threshold)
   disp("Signal 4: phrase detected");
else
   disp("Signal 4: phrase not detected");
end
```



Name: basil khowaja and yahya faraz

Correlated outputs:




```
threshold = 2.3188
Signal 1: phrase detected
Signal 2: phrase detected
Signal 3: phrase not detected


Signal 4: phrase detected
```

Name: basil khowaja and yahya faraz

Observation: The threshold is the lowest value of the correlation peaks of all the samples that are provided. As a result, any sound that is provided as input can be used to identify the "Okay" sound. This is not the best audio detector because it only has data from the four provided samples, making it difficult for it to detect any further samples. Its processing time will also increase as the dataset grows.