$$v_k(t) = Ae^{j(2\pi k f_o t + \phi)} \quad \text{(Eqn. 0)}$$

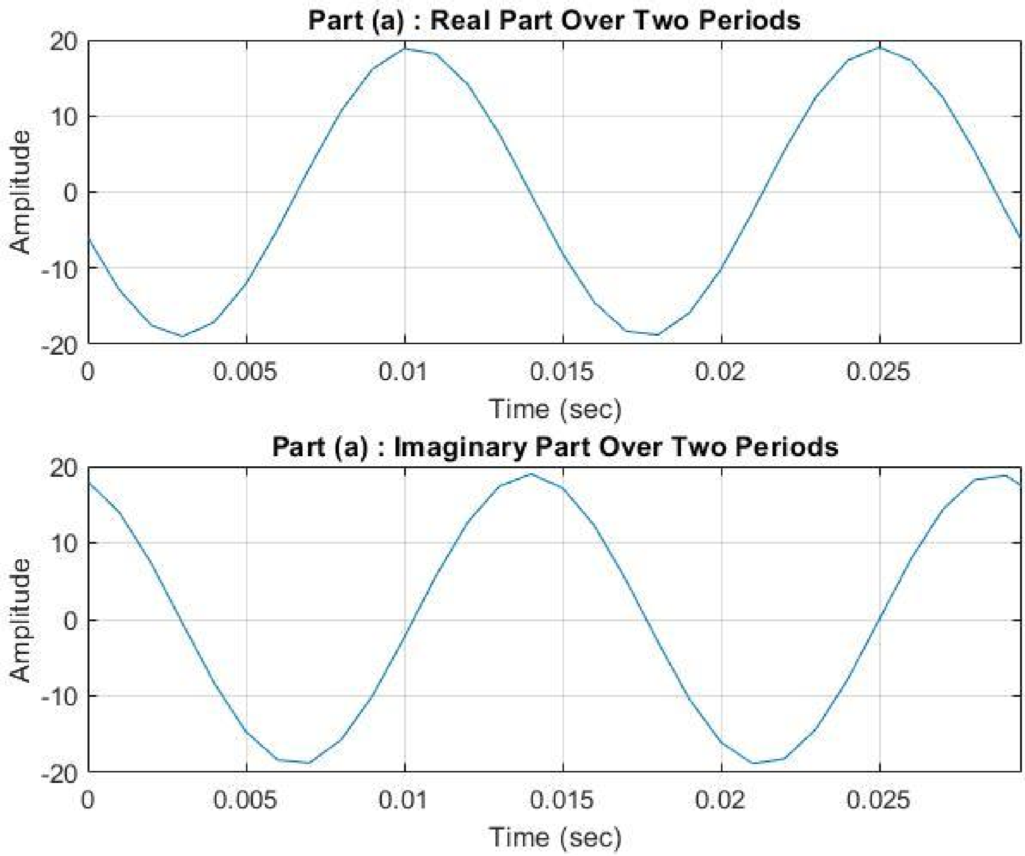
a) Using my ID number 21903419, A = 19, f_0=034=34 and I have chosen T_s as 10^{-3} from the set [. . . , 10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1]. Moreover, the rand() (I have calculated phase as phase = -pi + 2*pi*rand()) function gave me a phase value of 1.8867, and k = 2. From the equation and by the Euler's formula, the real and imaginary parts of the given signal can be found as:

$$x(t) = Re\{z(t)\} = Re\{Ae^{j(2\pi k f_o t+\phi)}\} = Acos(2\pi k f_o t + \phi)$$
$$y(t) = Im\{z(t)\} = Re\{Ae^{j(2\pi k f_o t+\phi)}\} = Asin(2\pi k f_o t + \phi)$$

(Eqn. 1&2)

By plugging the values into the equations above, and writing a code in MATLAB, I have plotted the real and imaginary parts of the signal over two fundamental periods as below:

A = 19, k = 2, f_0= 034, T_s = 10^{-1}, ϕ = 1.8867



(Graph 1&2)

b) In this part, I have kept the fundamental frequency, amplitude and phase the same, and chosen four different sampling interval T_s. I have also printed different parameters such as radian frequency, the digital frequency, the period of the signal, and the time shift Δt corresponding to the phase shift φ.

$$Radian\ Frequency\ \omega_0 = 2\pi f_0$$

$$Digital\ Frequency\ \widehat{\omega} = 2\omega T_s$$

$$Period\ of\ the\ Signal\ T_0 = 1/f_0$$

$$Time\ Shift\ \Delta t = \phi/(\omega_0)$$

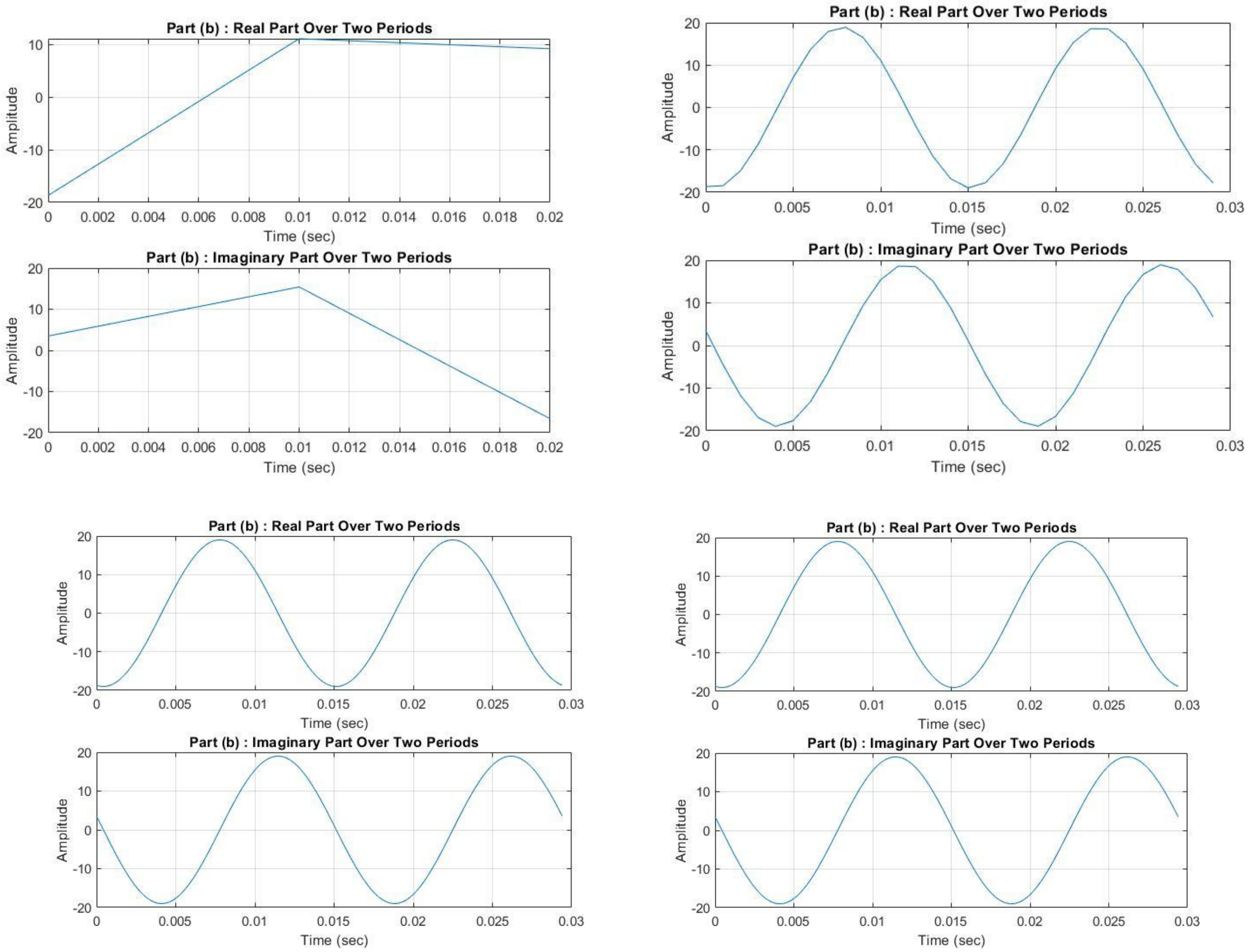
(Eqn. 3, 4, 5&6)

This time, phase value generated by rand() was 2.9568 and I have used four different T_s that are S = $\{10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}\}$.

```
Command Window
>> ma1_bsy
For T_s : 0.010000
Sequence : 19*exp(j*(2.136*k*n+2.957)
Radian frequency : 213.628
Digital Frequency : 2.136
Period of the signal : 0.029
Time shift : -0.014
-----------------------------------
For T_s : 0.001000
Sequence : 19*exp(j*(0.214*k*n+2.957)
Radian frequency : 213.628
Digital Frequency : 0.214
Period of the signal : 0.029
Time shift : -0.014
-----------------------------------
For T_s : 0.000100
Sequence : 19*exp(j*(0.021*k*n+2.957)
Radian frequency : 213.628
Digital Frequency : 0.021
Period of the signal : 0.029
Time shift : -0.014
-----------------------------------
For T_s : 0.000010
Sequence : 19*exp(j*(0.002*k*n+2.957)
Radian frequency : 213.628
Digital Frequency : 0.002
Period of the signal : 0.029
Time shift : -0.014
-----------------------------------
fx >>
```

(Figure 1: Sequence and variable values for different T_s values)

I have observed that graph of the signal gets smoother since we sample the signal more and more as we decrease the T_s value and taken more samples from the signal discretely. I have again sampled the signal over two periods. Moreover, as the T_s decreases and there are more samples from the signal, the digital frequency decreases as expected from equation 4. As we decrease T_s, f_s increases and we estimate the continuous-time complex signal as a result, as can be seen from the graphs below.



(Graphs 3-10)

c) I wrote a code to calculate the sum of real parts over a fundamental period, ideally, the result should have been 0; however, since the graph was plotted by sampling, it cannot be exactly continuous while being represented. The results obtained are generally between -2 and 2 for a T_s value of 0.0001. It can be said that this is one of the best-fit sampling rates for this signal because for the T_s values more or less than this value resulted in higher summations. This might be a result of errors that have arisen due to oversampling or undersampling the signal. I have repeated the same calculations for imaginary parts and results for both parts can be seen in the figure below.

```
Command Window
>> ma1_bsy
Sum of Real Part: -0.565490, Sum of Imag Part: -0.964129
fx >>
```

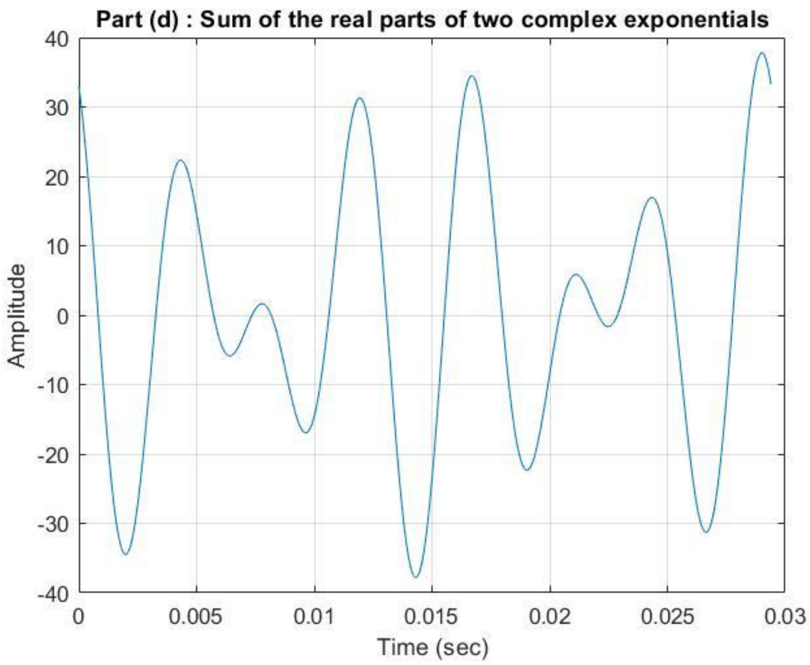
(Figure 2: Sums over a fundamental period)

d) I have created two signal variables with two different harmonics of the same fundamental frequency, this time the fundamental frequency is gcd of 170 and 238, which is still 34. I have set the variables to the values that can be seen in the figure below.

```
Command Window
>> ma1_bsy
A : 19, Phase : 0.521419
Fundamental Frequency : 34, Sampling Interval : 0.0001
fx >>
```

(Figure 3: Parameters for part d)

By adding up the real parts of the signals with k-values 5 and 7, I have obtained the third signal and plotted it over one fundamental period.



(Graph 11)

Then I have found the sum of the sum signal over 1, 3, and 5 fundamental periods. I have observed that I obtain negative values and it decreases as I increase the number of fundamental periods. For these values, it does not look like it is converging to a value.

```
>> ma1_bsy
Sum for 1 fundamental period:29.203423
Sum for 3 fundamental period:21.601030
Sum for 5 fundamental period:13.859720
>>
```

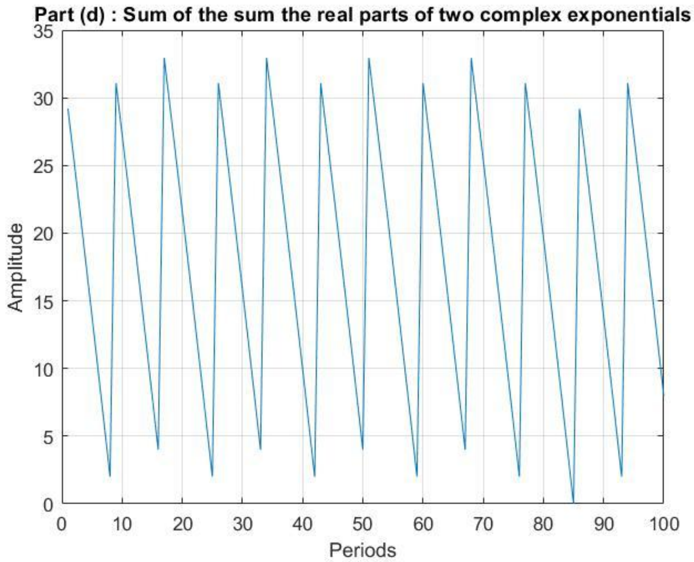
Figure 4: Sum over different periods.

However, I have observed that the sum values are repeating for every integer multiple of 17. For example sum over 1 period is the same as the sum over 18 periods. Moreover, I have also observed that the sum is 0 at every n*17 where n is a non-negative integer. When I tried to change my f_0 value, I have observed that the repeating period number is the half of fundamental frequency. In my case f_0 = 34 and it repeats for every 17 periods, I have also observed that sum of sum values takes values between 0 and my fundamental frequency..

```
Sum for 1 fundamental period:29.203423
Sum for 2 fundamental period:25.420038
Sum for 3 fundamental period:21.601030
Sum for 4 fundamental period:17.747290
Sum for 5 fundamental period:13.859720
Sum for 6 fundamental period:9.939228
Sum for 7 fundamental period:5.986731
Sum for 8 fundamental period:2.003153
Sum for 9 fundamental period:31.081481
Sum for 10 fundamental period:27.316239
Sum for 11 fundamental period:23.514931
Sum for 12 fundamental period:19.678445
Sum for 13 fundamental period:15.807678
Sum for 14 fundamental period:11.903532
Sum for 15 fundamental period:7.966923
Sum for 16 fundamental period:3.998769
Sum for 17 fundamental period:32.950303
Sum for 18 fundamental period:29.203423
```

Figure 5: Sum over 1 to 18 periods

Then to understand why, I have changed the fundamental periods from 1 to 100, and saved each value of the sum of the sum values to an array. Then I have plotted the real part of this array. I have observed that this graph is somewhat periodic and gets th at every 17 periods.



(Graph 12: y: amplitude, x: periods)

e) The inner product of two signals can be found by the equation:

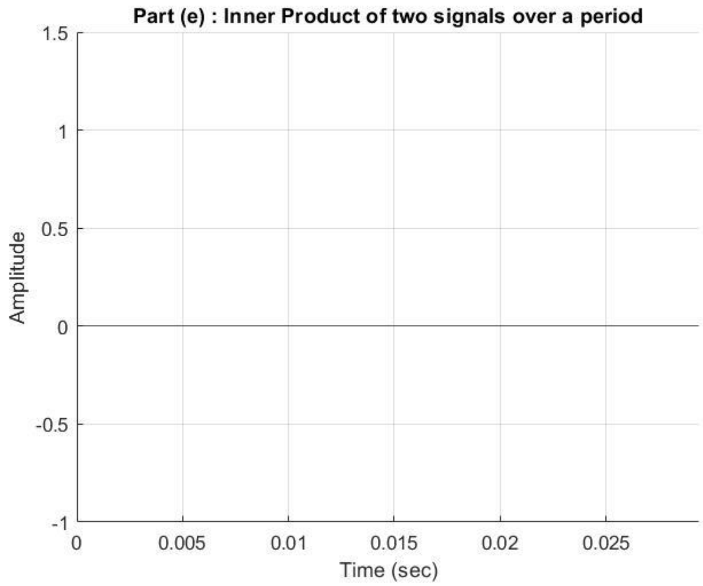
$$< v_k(t), v_l(t) > \ = \int_0^{T_0} (v_k(t) v_l^*(t))dt$$

(Eqn. 7)

I have obtained 0 for both imaginary and real parts as I expected from the inner product formula of two harmonics.

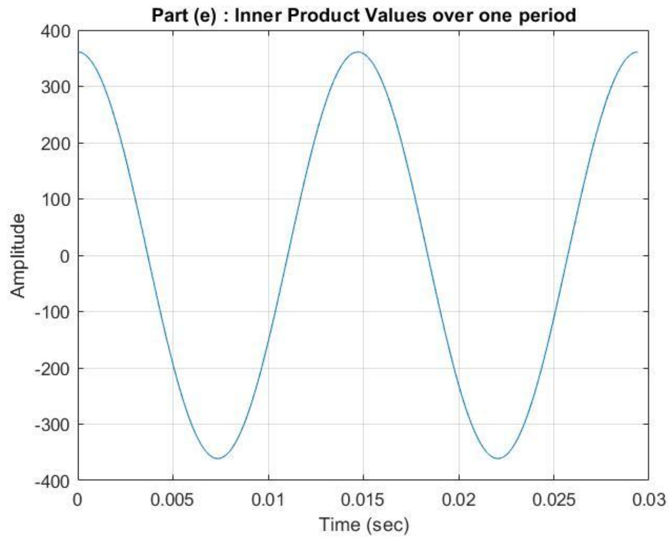
```
>> ma1_bsy
For harmonics 5 and 7: 0.000000 0.000000i
fx >>
```

(Figure 6: Result of inner product for harmonics 5 and 7)



(Graph 13: Scalar value graph)

Then I have plotted the inner product function which is the multiple of first signal with the conjugate of the second signal, and plotted it over one fundamental period:



(Graph 14)

I have observed that when I do not include the amplitude values to the multiplication, this signal oscillates between -1 and 1, now since it is multiplied with A*A, it oscillates between 361 and -361. Then I have calculated the sum of these values over one, three and five periods.

```
>> ma1_bsy
Sum for 1 fundamental period:318.533771
Sum for 3 fundamental period:233.591924
Sum for 5 fundamental period:148.644711
fx >>
```

(Figure 7: Sum over different periods)

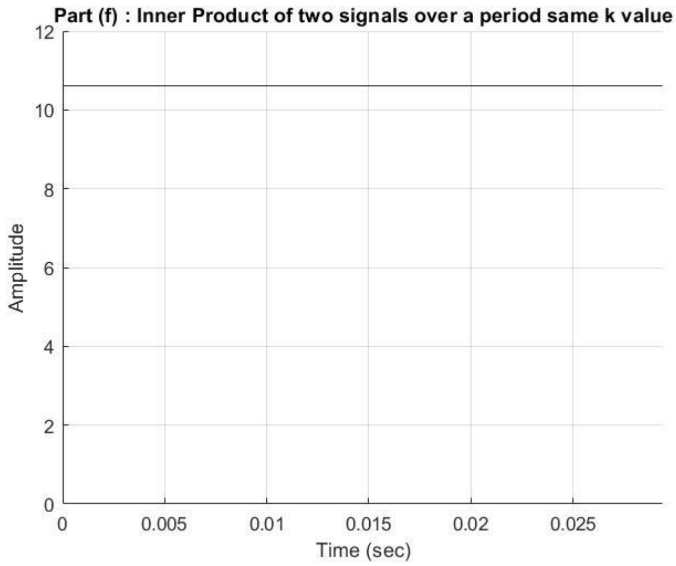
I have calculated the first 100 sums, and plotted them and realized that it was a similar graph to the one in part (d), but it was taking values between zero and 361, which is A*A, if I would divide sum values by A*A, it would oscillate between 0 and 1.

As I expected results were complying with orthogonality principle. In Fourier analysis we are trying to change the domain of signals, we try to go from time to frequency, and complex signals with different harmonics are periodic with T_0. By using this periodicity, we can represent Fourier Series coefficients, this is why we use complex signals.

f) As I tried in part d when I choose the same k values for two signals (fifth harmonic in this case), I obtain the following value by integrating.

```
>> ma1_bsy
For harmonics 5 and 5: 10.617647 0.000000i
>>
```

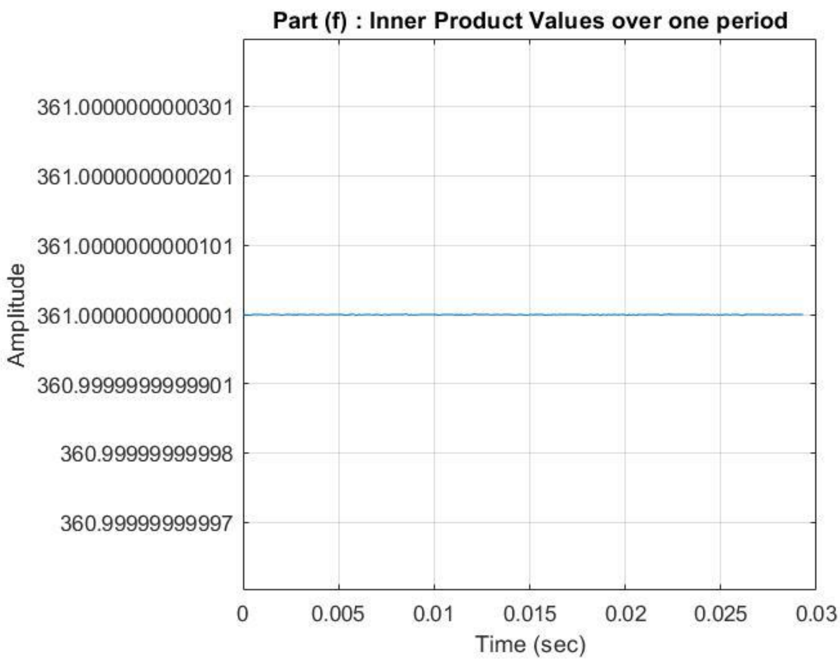
(Figure 8: Inner Product of two complex signals (integral result))



(Graph 14)

As orthogonality principle implies, for the same k value, the result of inner product is equal to T_0 value; however, it should be noted that there are A values to consider. Hence, the result of the inner product equals A*A* T_0 which is equal to the resulting value.

Then I have plotted the inner product function over one period:



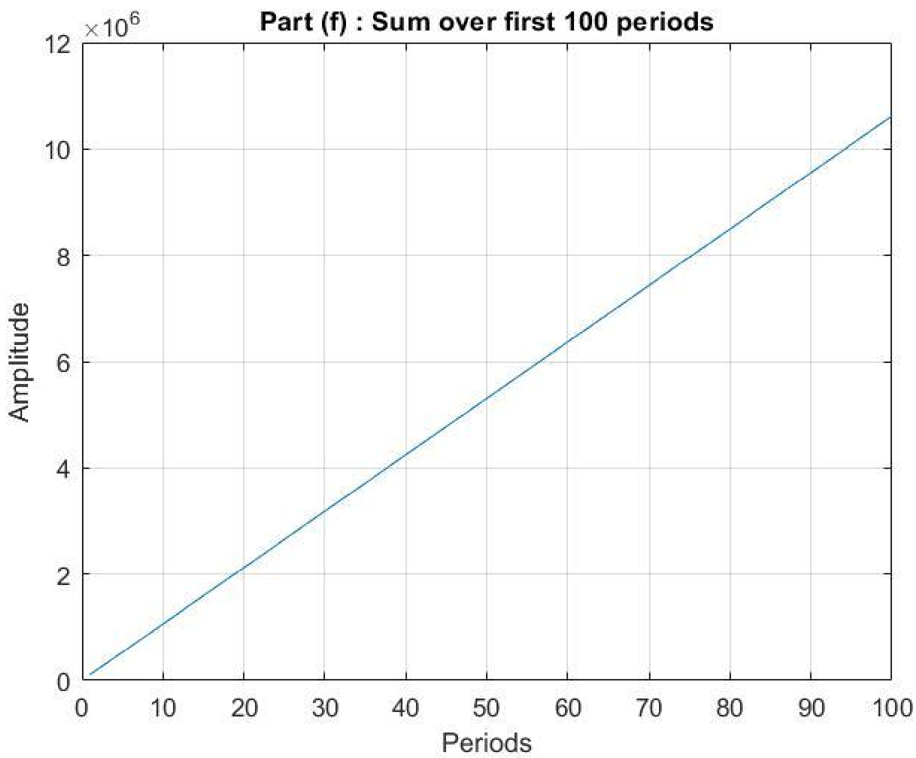
(Graph 15)

As in our textbook, the result of this function for same harmonics is 1, and in the graph it is 361 because it is multiplied with A*A. Since this is a constant value we cannot see a pattern as in part (e). Moreover, when I calculate the sum over 1, 3 and 5 periods, I have obtained:

```
>> ma1_bsy
Sum for 1 fundamental period:106495.000000
Sum for 3 fundamental period:318763.000000
Sum for 5 fundamental period:531031.000000
fx >>
```

(Figure 9: Sum over different periods)

Moreover, when I do the same calculation for sum over first 100 periods I have observed there is not a pattern existing, the values are only increasing. I have plotted this to visualize better:



(Graph 16)

Even if there are no repeating signals or oscillations, results are comply to the orthogonality principle.

MATLAB Code

Code for (a):

```
%% Author: Berk Saltuk Yılmaz
%% MATLAB Mini Project 1

% Part (a)
phase = -pi + 2*pi*rand();

A = 19; % Last 2 digits of 21903419
f_0 = 34; % Last 3 digits before last 2 digits 034; used as 34
k = 2; % Given in part a

T_s = 0.001; % T_s is 10^-3

t = 0:0.001:1/f_0+0.001; % Ts = 10^-1


x = A*exp(1i*(2*pi*f_0*k.*t+phase));

r_1 = A*cos(2*pi*f_0*k.*t+phase);

im_1 = A*sin(2*pi*f_0*k.*t+phase);

figure;
subplot (2,1,1);
plot(t,r_1);
title('Part (a) : Real Part Over Two Periods');
grid on;
xlabel('Time (sec)')
ylabel('Amplitude')

subplot (2,1,2);
plot(t,im_1);
title('Part (a) : Imaginary Part Over Two Periods');
grid on;
xlabel('Time (sec)')
ylabel('Amplitude')
```

Code for (b):

```
% Part B
phase = -pi + 2*pi*rand();

T_s = 0.01; % Sampling Interval
```

```
plotSequenceAndPrint(T_s,phase);

T_s = 0.001;
plotSequenceAndPrint(T_s,phase);

T_s = 0.0001;
plotSequenceAndPrint(T_s,phase);

T_s = 0.00001;
plotSequenceAndPrint(T_s,phase);

function plotSequenceAndPrint(T_s, phase)

    A = 19; % Last 2 digits of 21903419
    f_0 = 34; % Last 3 digits before last 2 digits 034; used as 34
    k = 2; % Given in part a

    t = 0:T_s:1/(f_0 + 0.001);

    real = A*cos(2*pi*f_0*k.*t+phase);
    imag = A*sin(2*pi*f_0*k.*t+phase);

    figure;
    subplot (2,1,1);
    plot(t, real);
    title('Part (b) : Real Part Over Two Periods');
    grid on;
    xlabel('Time (sec)')
    ylabel('Amplitude')

    subplot (2,1,2);
    plot(t, imag);
    title('Part (b) : Imaginary Part Over Two Periods');
    grid on;
    xlabel('Time (sec)')
    ylabel('Amplitude')

    w_0 = 2*pi*f_0; % Radian Frequency
    w_hat = w_0*T_s; % Digital Frequency
    T_0 = 1/f_0; % Period of the signal
    delta_t = -1*(phase/w_0); % Time shift corresponding to the phase
shift
    fprintf("For T_s : %f\n", T_s);
    fprintf("Sequence : 19*exp(j*(%.3f*k*n+%.3f)\n", w_hat,phase);
    fprintf("Radian frequency : %.3f\n", w_0);
```

```
    fprintf("Digital Frequency : %.3f\n", w_hat);
    fprintf("Period of the signal : %.3f\n", T_0);
    fprintf("Time shift : %.3f\n", delta_t);
    fprintf("------------------------------------\n");
end
```

Code for (c):

```
% Part C
phase = -pi + 2*pi*rand();

A = 19; % Last 2 digits of 21903419
f_0 = 34; % Last 3 digits before last 2 digits 034; used as 34
k = 2; % Given in part a

T_s = 0.0001; % T_s is 10^-3
t = 0:T_s:(1/(2*f_0)) - T_s; % Ts = 10^-3

im_1 = A*sin(2*pi*f_0*k*t+phase);
sumRe = 0;
sumIm = 0;

for t_vals = 1 : length(t)
    sumRe = sumRe + A*cos((2*pi*f_0*k*t(t_vals)) + phase);
    sumIm = sumIm + A*sin((2*pi*f_0*k*t(t_vals)) + phase);
end

fprintf("Sum of Real Part: %f, Sum of Imag Part: %f\n", sumRe, sumIm);
```

Code for (d):

```
%Part D
phase = 0.521419;
A = 19; % Last 2 digits of 21903419
f_0 = 34; % Last 3 digits before last 2 digits 034; used as 34
T_0 = 1/f_0;

T_s = 0.0001; % T_s is 10^-3
t = 0:T_s:T_0 - T_s; % Ts = 10^-1

signal1 = A*exp(1i*((2*pi*f_0*5*t)+phase)); % K = 5
signal2 = A*exp(1i*((2*pi*f_0*7*t)+phase)); % K = 7

fprintf("A : %d, Phase : %f\nFundamental Frequency : %d, Sampling
Interval : %.3f\n", A, phase, f_0, T_s);
```

```
newSignal = signal1 + signal2;
figure;

plot(t, real(newSignal));
title('Part (d) : Sum of the real parts of two complex exponentials');
grid on;
xlabel('Time (sec)')
ylabel('Amplitude')

set = [1, 3, 5];
T_s = 0.0001;
for index = 1 : length(set)
    T = set(index)/(f_0);
    increment= 0;
    sum_harmonics = 0;
    while increment < T
        sum_harmonics = sum_harmonics +
A*exp(1i*((2*pi*f_0*5*increment)+phase)) +
(A*exp(-1i*((2*pi*f_0*7*increment)+phase)));
        increment = increment + T_s;
    end
    fprintf("Sum for %d fundamental period:%f\n", set(index),
sum_harmonics);
end
set = 1:1:100;
T_s = 0.0001;
x = [];
for index = 1 : length(set)
    T = set(index)/(f_0);
    increment= 0;
    sum_harmonics = 0;
    while increment < T
        sum_harmonics = sum_harmonics +
A*exp(1i*((2*pi*f_0*5*increment)+phase)) +
(A*exp(-1i*((2*pi*f_0*7*increment)+phase)));
        increment = increment + T_s;
    end
    fprintf("Sum for %d fundamental period:%f\n", set(index),
sum_harmonics);
    x(index) = sum_harmonics;
end

plot(set, x)
```

Code for (e):

```
%Part E
phase = 0;
A = 19; % Last 2 digits of 21903419
f_0 = 34; % Last 3 digits before last 2 digits 034; used as 34
T_0 = 1/f_0;

syms t_param

func =
(A*exp(1i*((2*pi*f_0*5*t_param)+phase))).*conj((A*exp(1i*((2*pi*f_0*7*t_
param)+phase))));

inn_product = int(func, 0, T_0);

fprintf('For harmonics 5 and 7: %f %fi\n',
real(inn_product),imag(inn_product));
T_s = 0.0001;
t = 0:T_s:T_0 - T_s;
innerproduct_signal =
A*exp(1i*((2*pi*f_0*5*t)+phase)).*conj((A*exp(1i*((2*pi*f_0*7*t)+phase))
));
plot(t, real(innerproduct_signal))
title('Part (e) : Inner Product Values over one period');
grid on;
xlabel('Time (sec)')
ylabel('Amplitude')

set = [1,3,5];
for index = 1 : length(set)
    T = set(index)/(f_0);
    increment= 0;
    sum_inner = 0;
    while increment < T
        sum_inner = sum_inner +
A*exp(1i*((2*pi*f_0*5*increment)+phase)).*(A*exp(-1i*((2*pi*f_0*7*increm
ent)+phase)));
        increment = increment + T_s;
    end
    fprintf("Sum for %d fundamental period:%f\n", set(index),
sum_inner);
end
set = 1:1:100;
T_s = 0.0001;
plotthis = [];
```

```
for index = 1 : length(set)
    T = set(index)/(f_0);
    increment= 0;
    sum_inner = 0;
    while increment < T
        sum_inner = sum_inner +
A*exp(1i*((2*pi*f_0*5*increment)+phase)).*(A*exp(-1i*((2*pi*f_0*7*increm
ent)+phase)));
        increment = increment + T_s;
    end
    fprintf("Sum for %d fundamental period:%f\n", set(index),
sum_inner);
    plotthis(index) = sum_inner;
end

plot(set, plotthis);


yline(real(inner_product))
title('Part (e) : Inner Product of two signals over a period');
grid on;
xlabel('Time (sec)')
ylabel('Amplitude')
xlim([0 T_0])
```

Code for (f):

```
%Part F
phase = 0;
A = 19; % Last 2 digits of 21903419
f_0 = 34; % Last 3 digits before last 2 digits 034; used as 34
T_0 = 1/f_0;

syms t_param

func_to_integrate =
(A*exp(1i*((2*pi*f_0*5*t_param)+phase))).*conj((A*exp(1i*((2*pi*f_
0*5*t_param)+phase))));

inner_product = int(func_to_integrate, 0, T_0);

% fprintf('For harmonics 5 and 5: %f %fi\n',
real(inner_product),imag(inner_product));
```

```
T_s = 0.0001;
t = 0:T_s:T_0 - T_s;
innerproduct_signal =
A*exp(1i*((2*pi*f_0*5*t)+phase)).*conj((A*exp(1i*((2*pi*f_0*5*t)+p
hase))));
plot(t, real(innerproduct_signal))
title('Part (f) : Inner Product Values over one period');
grid on;
xlabel('Time (sec)')
ylabel('Amplitude')

set = [1,3,5];
for index = 1 : length(set)
    T = set(index)/(f_0);
    increment= 0;
    sum_inner = 0;
    while increment < T
        sum_inner = sum_inner +
A*exp(1i*((2*pi*f_0*5*increment)+phase)).*(A*exp(-1i*((2*pi*f_0*5*
increment)+phase)));
        increment = increment + T_s;
    end
    fprintf("Sum for %d fundamental period:%f\n", set(index),
sum_inner);
end
set = 1:1:100;
T_s = 0.0001;
plotthis = [];
for index = 1 : length(set)
    T = set(index)/(f_0);
    increment= 0;
    sum_inner = 0;
    while increment < T
        sum_inner = sum_inner +
A*exp(1i*((2*pi*f_0*5*increment)+phase)).*(A*exp(-1i*((2*pi*f_0*5*
increment)+phase)));
        increment = increment + T_s;
    end
    fprintf("Sum for %d fundamental period:%f\n", set(index),
sum_inner/A^2);
    plotthis(index) = sum_inner;
end
```

```
plot(set, plotthis);
title('Part (f) : Sum over first 100 periods');
grid on;
xlabel('Periods')
ylabel('Amplitude')
```