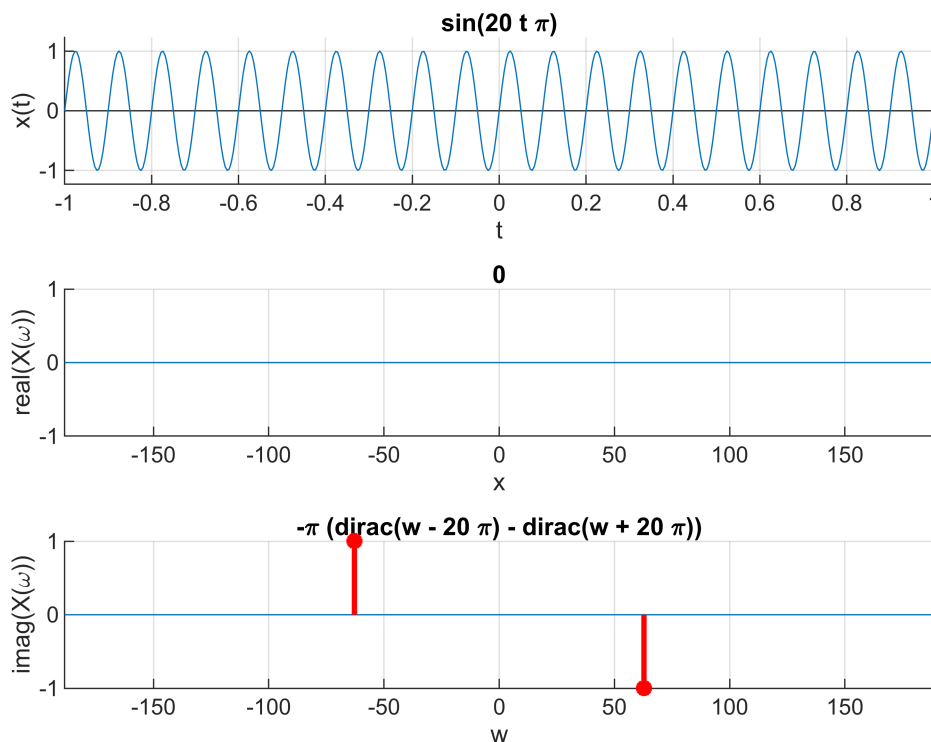


### Ćwiczenie 1

Transformata fouriera sygnału  $\sin(\omega t)$

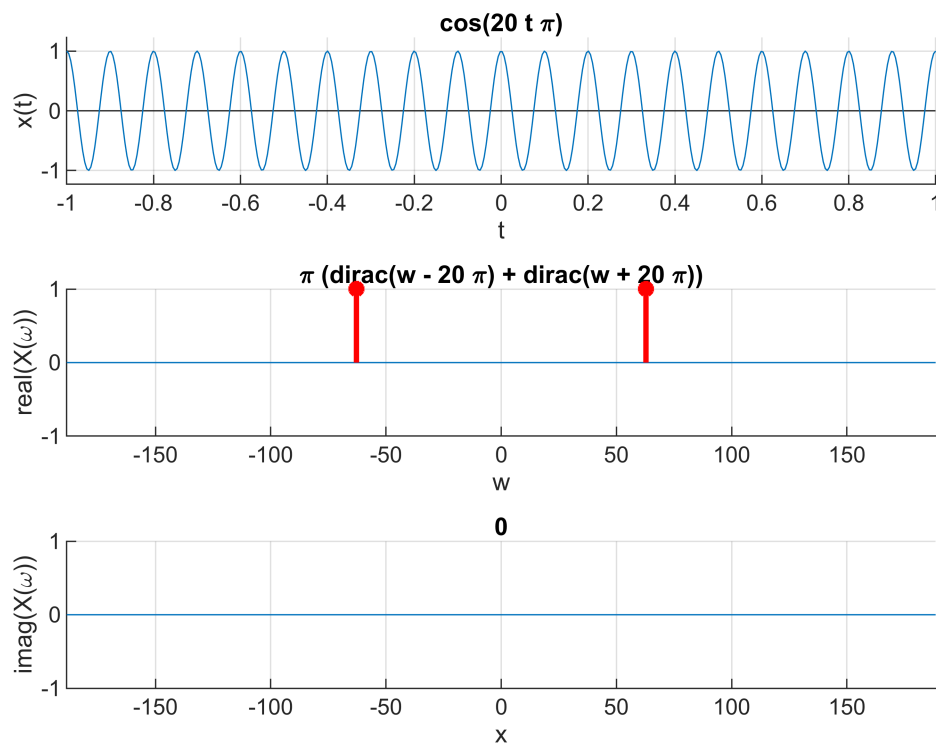
```
syms t
f0 = 10; %Hz
w0 = 2*pi*f0;
x = sin(w0*t);
fourT_dirac(x, f0)
```



### Zad 2

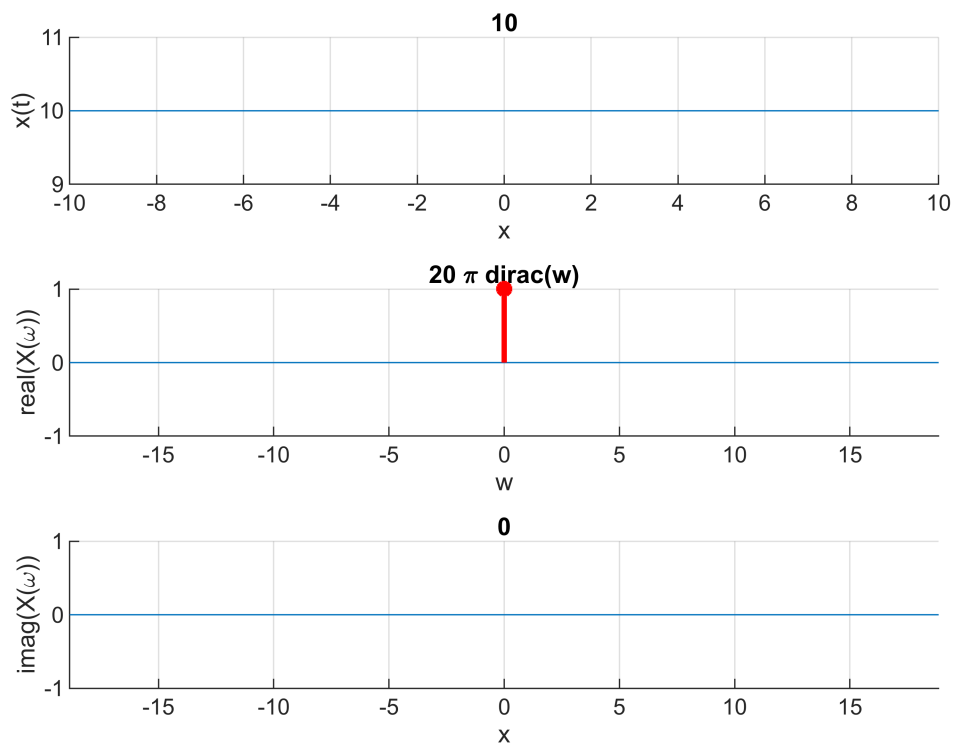
a) sygnał  $\cos(\omega_0 t)$

```
f0 = 10; %Hz
w0 = 2*pi*f0;
x = cos(w0*t);
fourT_dirac(x, f0)
```



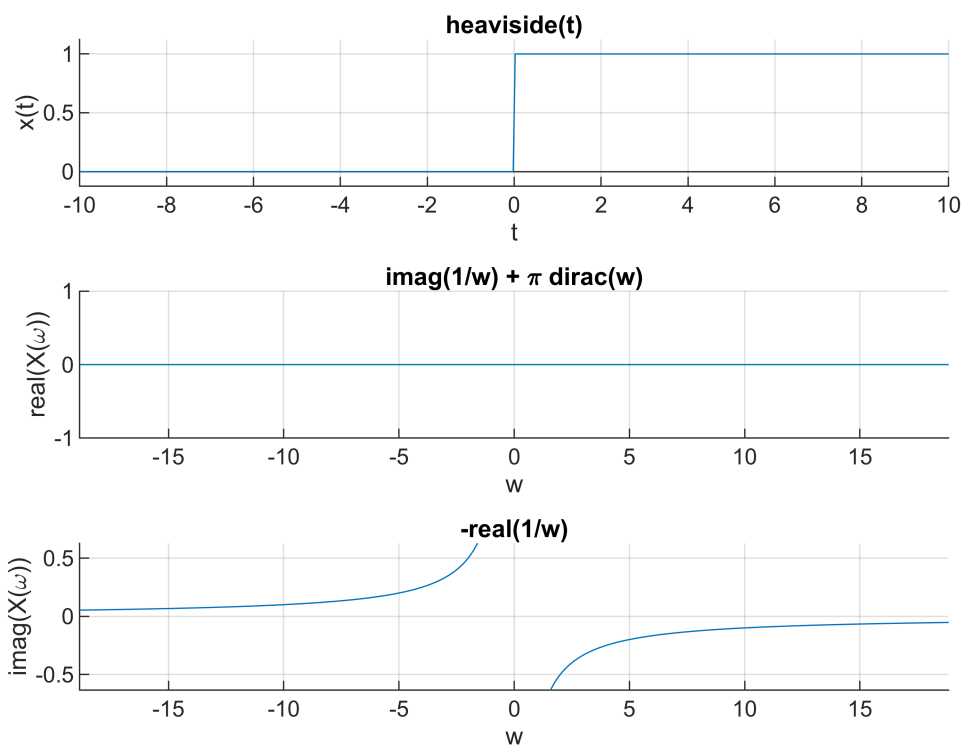
**b) sygnał stały o wartości 10 (do jego zdefiniowania należy użyć polecenia `sym(10)`)**

```
f0 = 1; %Hz
x = sym(10);
fourT_dirac(x, f0)
```



### c) skok jednostkowy o amplitudzie 1

```
f0 = 1; %Hz
x = heaviside(t);
fourT_cont(x, f0)
```

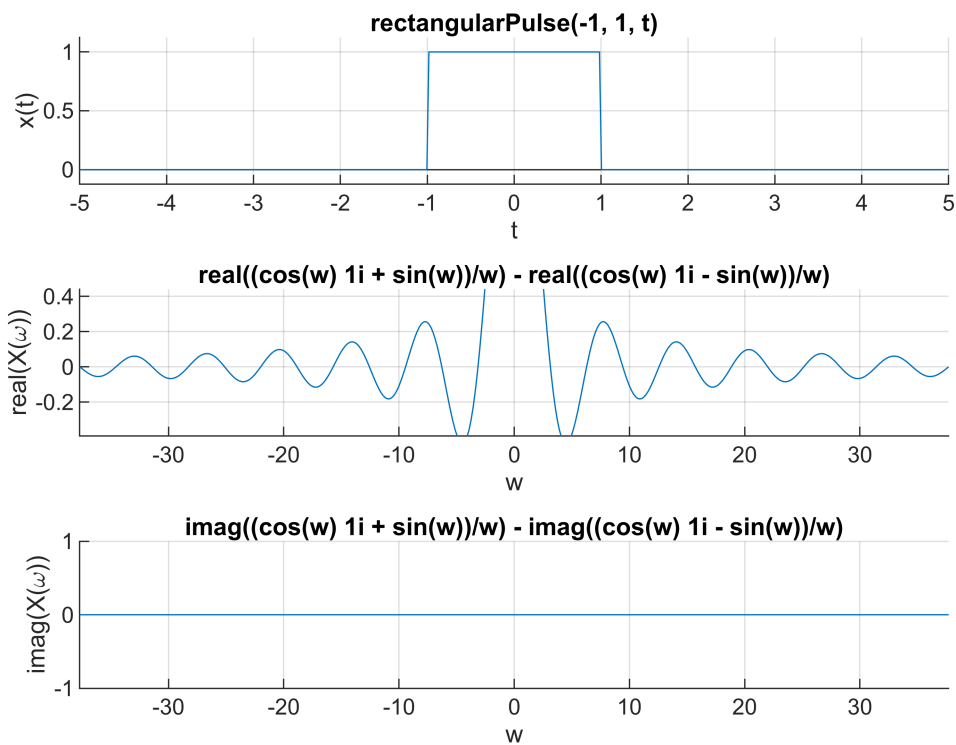


**d) impuls prostokątny o amplitudzie 1 oraz czasie trwania  $T_i = 2/f_0$**

```
f0 = 2;  
x = rectangularPulse(-1, 1, t)
```

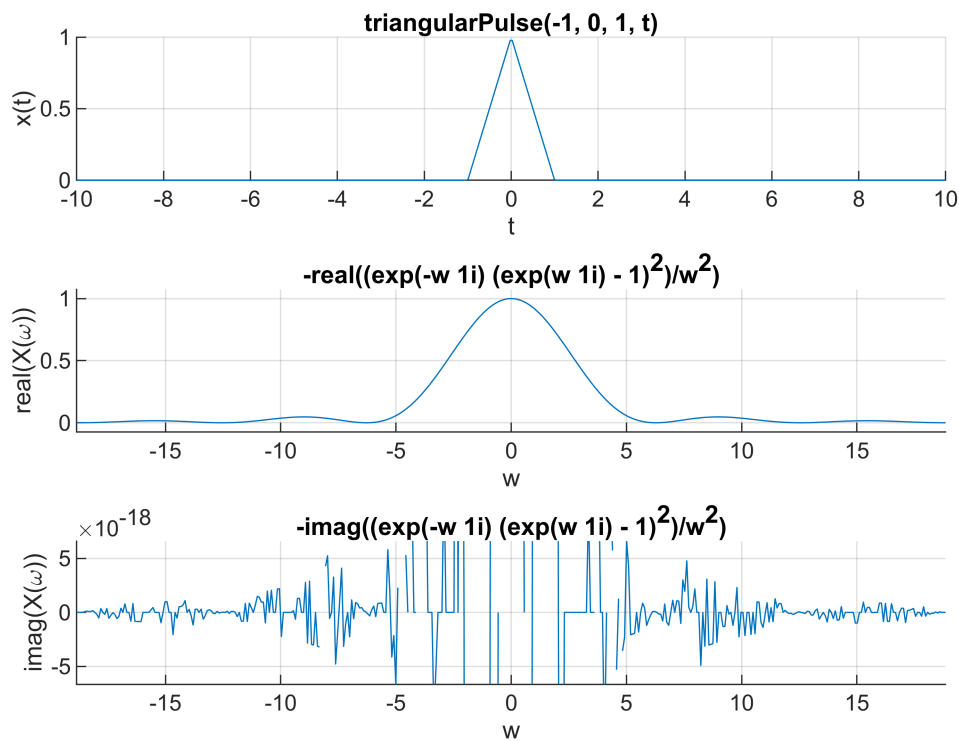
```
x = rectangularPulse(-1, 1, t)
```

```
fourT_cont(x, f0)
```



**e) impuls trójkątny (symetryczny) o amplitudzie 1 oraz czasie trwania  $T_i = 2/f_0$  .**

```
f0 = 1;
x = triangularPulse(-1/f0, 1/f0, t);
fourT_cont(x, f0)
```



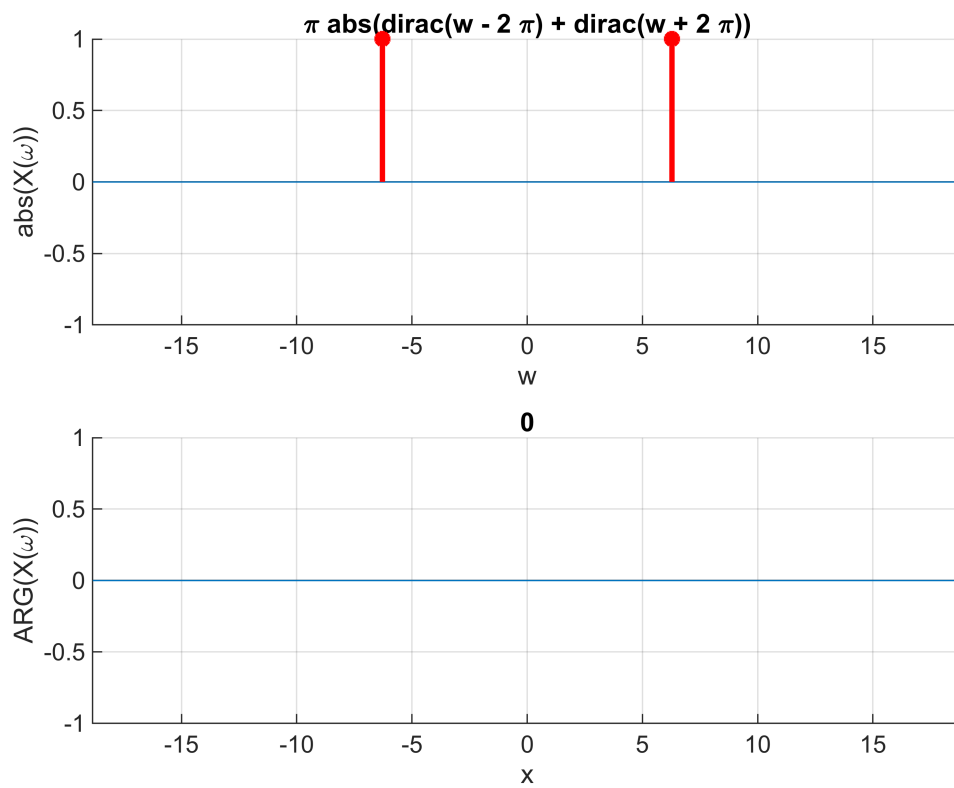
## ZD2

Dla przebiegów  $\cos(\omega_0 t)$  oraz  $\sin(\omega_0 t)$  sporządź również wykresy gęstości widmowej amplitudy oraz fazy. Porównaj je i opisz różnice w sprawozdaniu.

### Cos

```
syms w t
f0=1;
w0 = 2*pi* f0;
x = cos(w0*t);

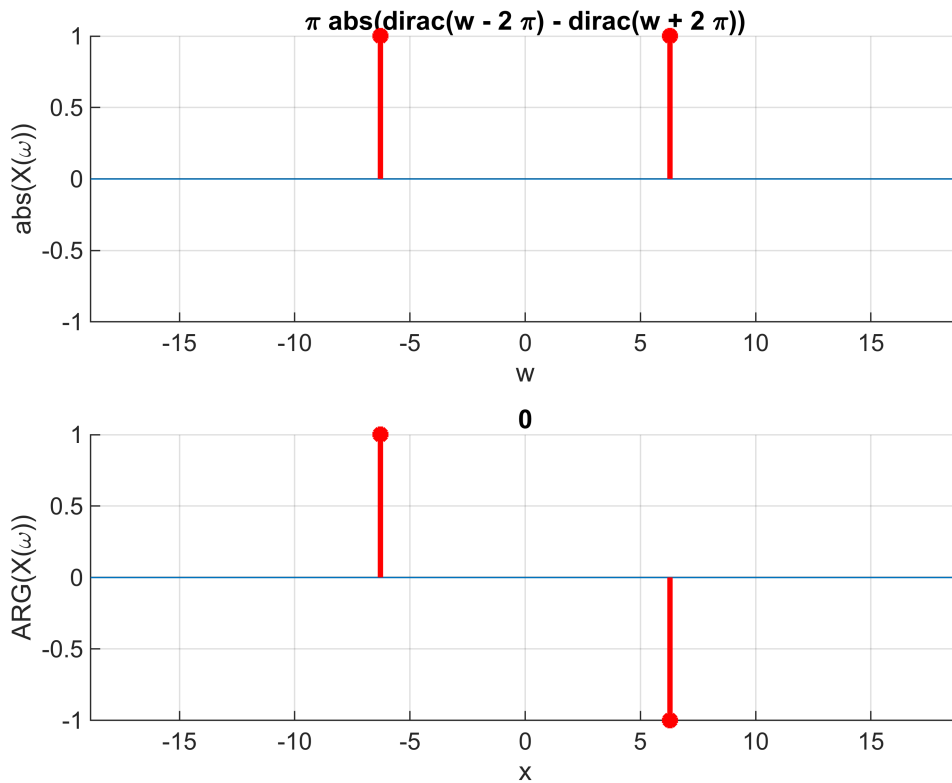
fourT_dirac_gestosc(x, f0)
```



## Sin

```
syms w t
f0=1;
w0 = 2*pi* f0;
x = sin(w0*t);

fourT_dirac_gestosc(x, f0)
```



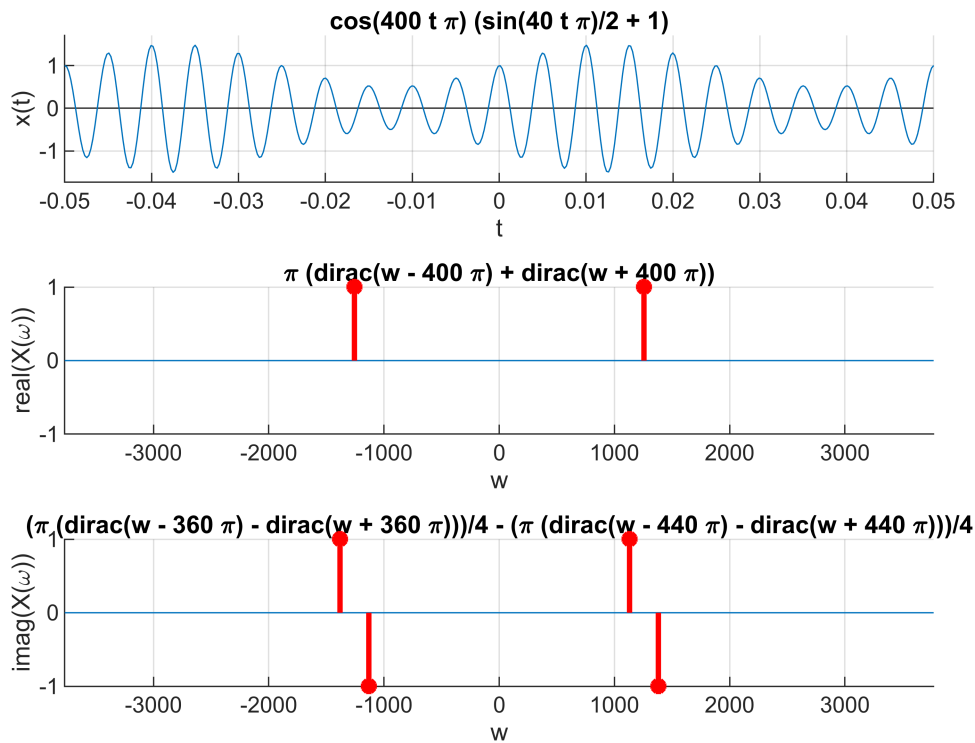
Sin i cos posiadają takie same amplitudy, cos ma fazę zerową a sin 2pi.

### Zadanie 3

Przeprowadź modulację sygnału  $\cos(\omega_0 t)$  przebiegiem sinusoidalnym o częstotliwości 10-krotnie mniejszej, amplitudzie 1 i głębokości modulacji  $m = 0.5$ . Wyznacz transformatę Fouriera sygnału zmodulowanego i uzgodnij skalę prędkości widma. Kod programu oraz wykresy samieść w sprawozdaniu

```
syms t
f0 = 1;
f1 = 10;
w0 = 2*pi*f0;
w1 = 2*pi*f1;
%modulacja amplitudowa
m = 0.5;
x = cos(w1*t) * (1 + m*sin(w0*t));
fourT_dirac(x, f1);
```





### ZD3

Porównaj moc sygnału nośnego z mocą wynikowego sygnału zmodulowanego. Napisz w sprawozdaniu jaki wpływ na tę relację ma głębokość modulacji  $m$ .

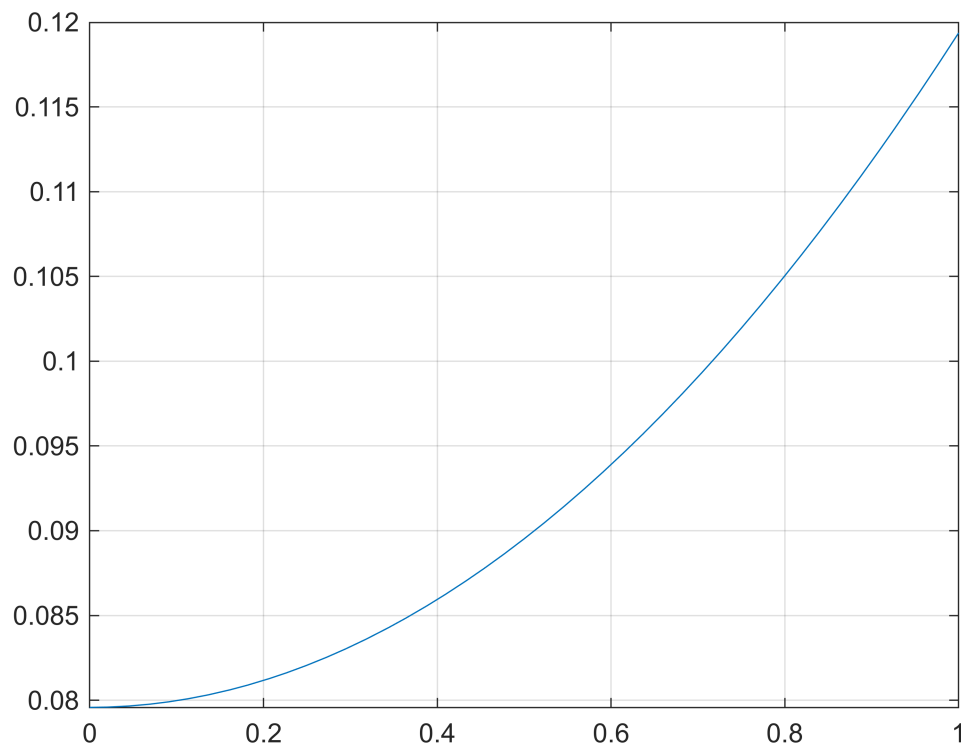
```
syms t w
m_v = 0:0.1:1;
power_v = m_v;
i = 1;
f1 = 10;
f0 = f1/10;
w0 = 2 * pi * f0;
w1 = 2 * pi * f1;

syms t m
x = cos(w1*t)*(1 + m*sin(w0*t));
px = 1/w0 * int(x^2, t, 0, 1/f0);
```

px =

$$\frac{5734161139222659 m^2}{144115188075855872} + \frac{5734161139222659}{72057594037927936}$$

```
figure
fplot(px, [0, 1]); grid on
```



Moc zwiększa się z kwadratem głębokości populacji do około 150% wartości mocy sygnału niezmodulowanego

## Zad 4

Wyznacz transformatę Fouriera sygnału  $\cos(10t)$  "okienkowanego" tak aby w oknie znajdowa

ło się 6.5 okresów przebiegu  $T_0 = 6.5 = f_0$ . Dzięki temu, przebieg rozpocznie się

w obrębie okna i zakończy wartości się wartości zerowej. Do okienkowania zastosuj

odpowiednio przeskalowane sygnały z Zad. 2:

```
syms t
f0 = 1;
```

```
f0 = 1
```

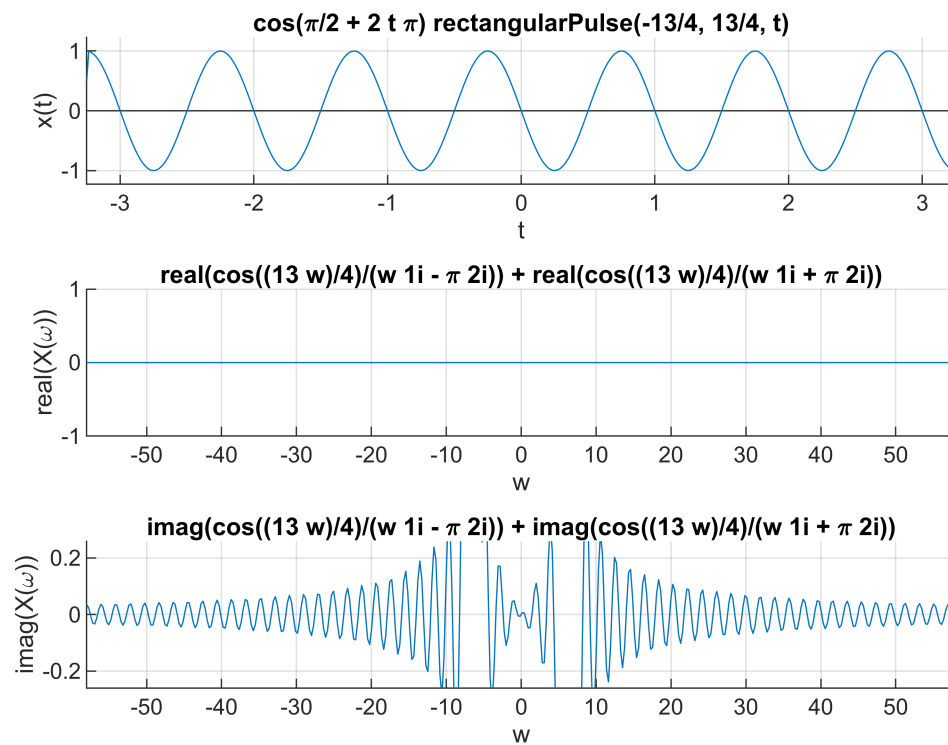
```
w0 = 2*pi*f0;
```

```
w0 = 6.2832
```

```
x = cos(w0*t+pi/2);
```

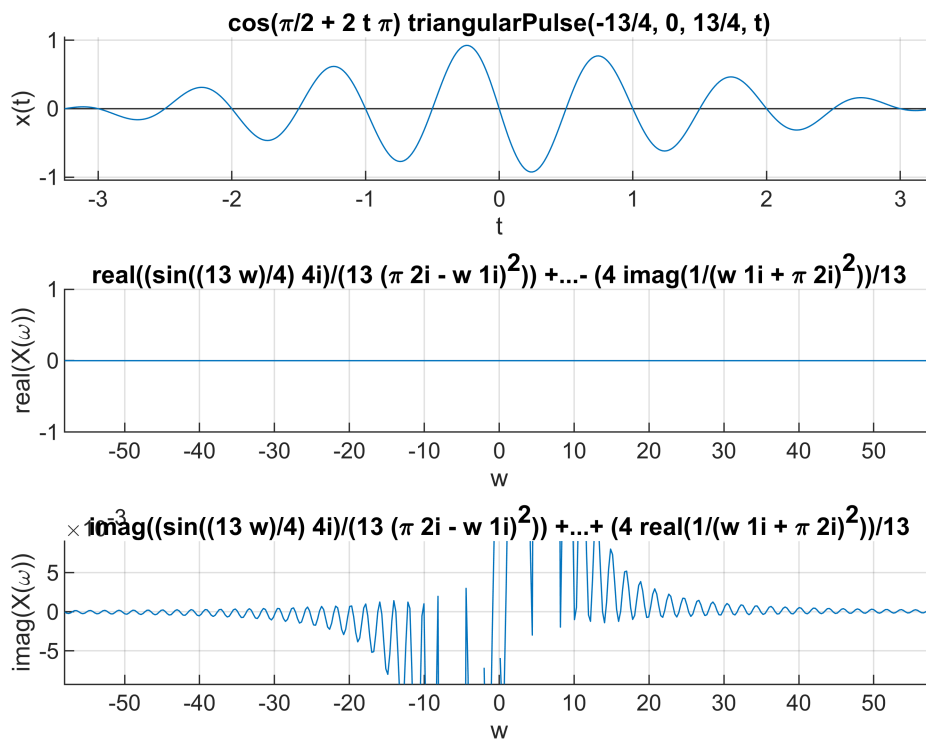
okienkowanie sygnałem prostokątnym

```
okno = rectangularPulse(-3.25, 3.25, t);
fourT_cont_o(x, okno, 3.25)
```



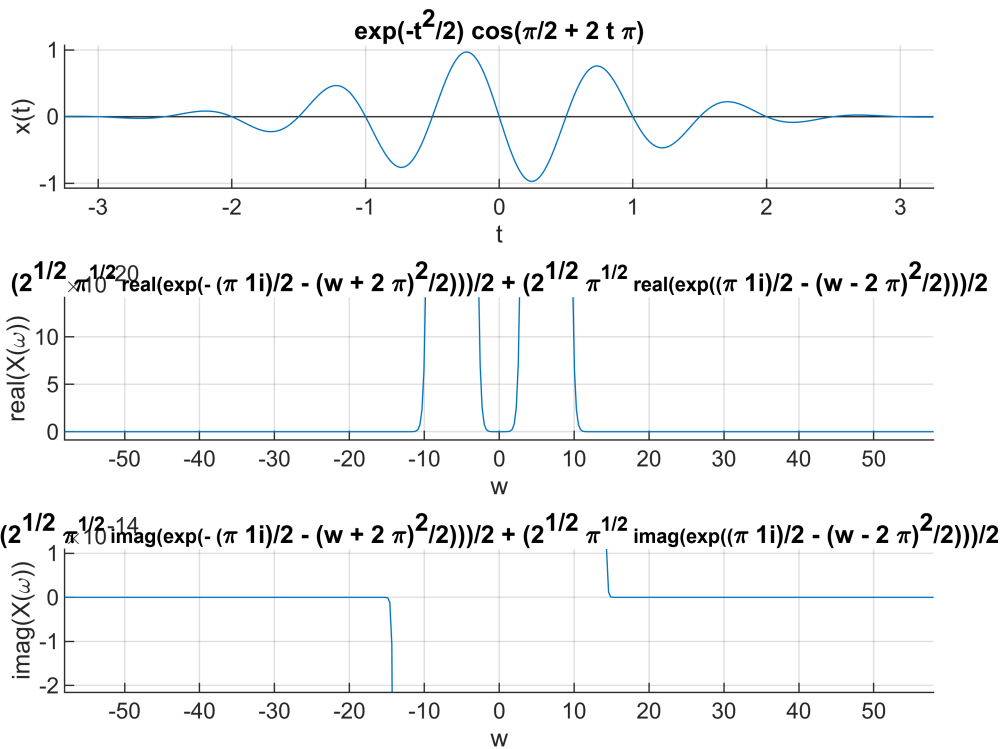
okienkowanie sygnałem trójkątnym

```
okno = triangularPulse(-3.25, 3.25, t);
fourT_cont_o(x, okno, 3.25)
```



okienkowanie sygnałem krzywej gaussa

```
okno = exp((-t^2)/2);
fourT_cont_o(x, okno, 3.25)
```



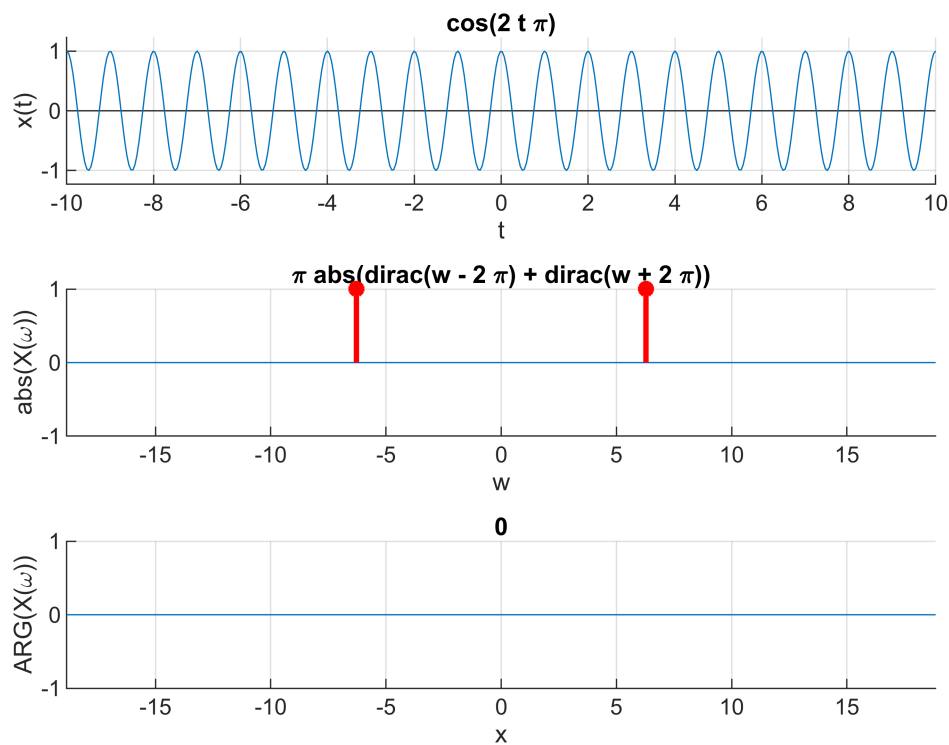
### ZD3

Dla każdego przypadku sporządź wykres gęstości widmowej amplitudy sygnału. porównaj uzyskane wyniki z widmem sygnału  $\cos(\omega_0 t)$  niepoddanego okienkowaniu. Wskaż najlepszą metodę okienkowania i uzasadnij wybór.

```
syms t
f0 = 1;
w0 = 2*pi*f0;
x = cos(w0*t);
```

nieokienkowany

```
fourT_dirac_gestosc(x, f0)
```

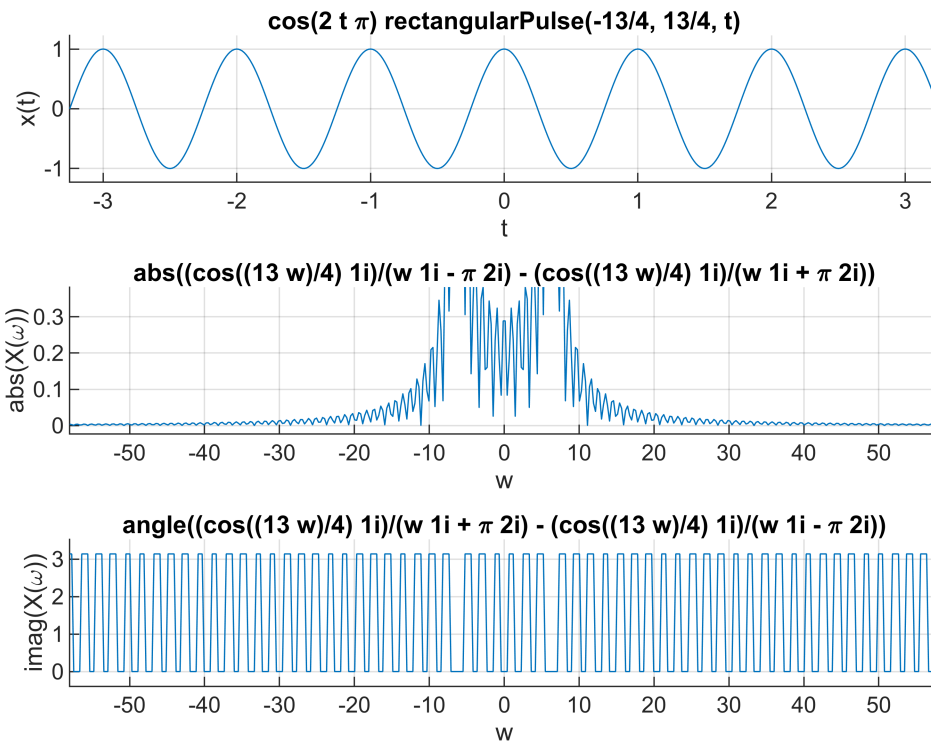


okienkowanie sygnałem prostokątnym

```
sz_okna = 6.5
```

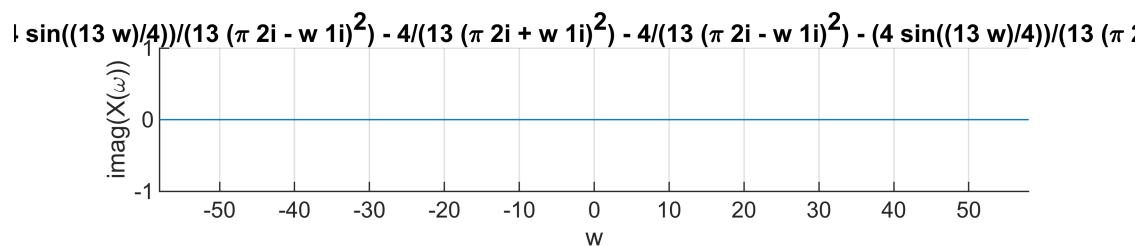
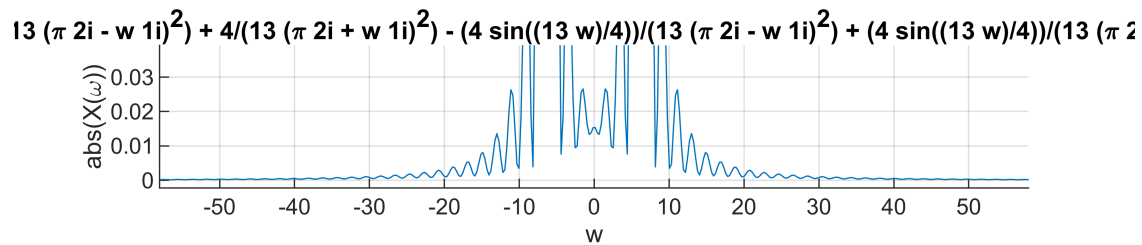
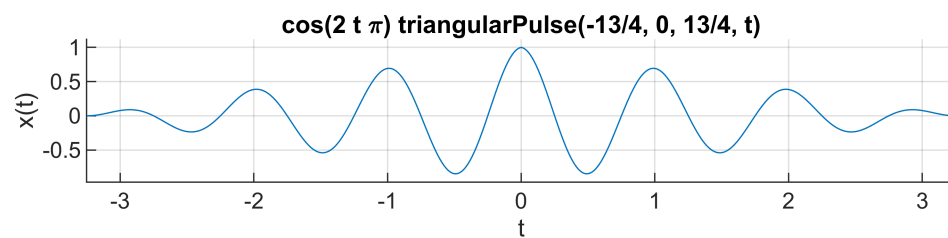
```
sz_okna = 6.5000
```

```
okno = rectangularPulse(-sz_okna/2, sz_okna/2, t);  
fourT_cont_o_gestosc(x, okno, sz_okna/2)
```



okienkowanie sygnałem trójkątnym

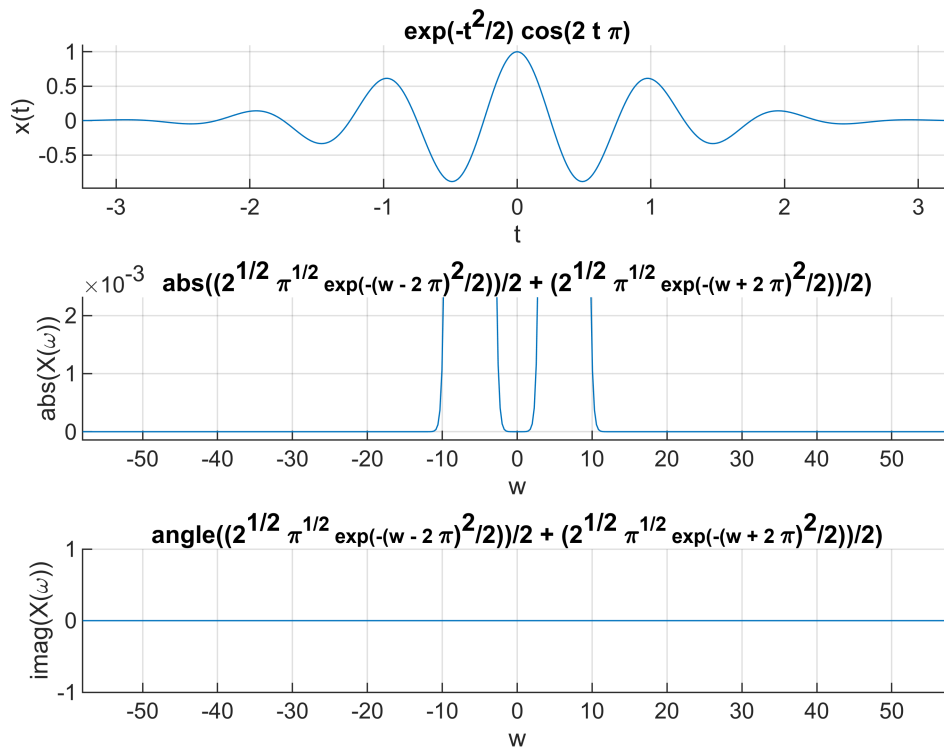
```
okno = triangularPulse(-sz_okna/2, sz_okna/2, t);
fourT_cont_o_gestosc(x, okno, sz_okna/2)
```



okienkowanie sygnałem krzywej gaussa

```
okno = exp((-t^2)/2);
fourT_cont_o_gestosc(x, okno, sz_okna/2)
```





Najmniej zaszumioną amplitudę i fazę daje okienkowanie krzywą Gaussa

```
function [] = fourT_dirac(x, f0)

syms w t

w0 = 2*pi*f0;
BND_t = [-10/f0;10/f0]; %20 okresow
t_SMP = [BND_t(1):1/(10*f0):BND_t(2)];
BND_w = [-3*w0;3*w0];
w_SMP = [BND_w(1):w0/10:BND_w(2)];
X_FT = fourier(x);

%plot signal
figure
subplot(3,1,1); ylabel('x(t)'); hold on
ezplot(x,BND_t); hold on; grid on;
v_num = subs(x, t, t_SMP);
n = find(abs(v_num) == inf); % plot dirac (inf)
stem(t_SMP(n),sign(v_num(n)), 'r*', 'LineWidth', 2);

%plot real of fourier
subplot(3,1,2); ylabel('real(X(\omega))'); hold on
ezplot(real(X_FT), BND_w); hold on; grid on;
v_num = subs(real(X_FT), w, w_SMP);
```

```

n = find( abs(v_num) == inf); % plot dirac (inf)
stem(w_SMP(n),sign(v_num(n)), 'r*', 'LineWidth', 2);

%plot imag of fourier
subplot(3,1,3); ylabel('imag(X(\omega))'); hold on
ezplot(imag(X_FT), BND_w); hold on; grid on
v_num = subs(imag(X_FT), w, w_SMP);
n = find( abs(v_num) == inf ); % plot dirac (inf)
stem(w_SMP(n),sign(v_num(n)), 'r*', 'LineWidth', 2);

end

function [] =fourT_dirac_gestosc(x, f0)

syms w t

w0 = 2*pi*f0;
BND_t = [-10/f0;10/f0]; %20 okresow
t_SMP = [BND_t(1):1/(10*f0):BND_t(2)];
BND_w = [-3*w0;3*w0];
w_SMP = [BND_w(1):w0/10:BND_w(2)];
X_FT = fourier(x);

%plot signal
figure
subplot(3,1,1); ylabel('x(t)'); hold on
ezplot(x,BND_t); hold on; grid on;
v_num = subs(x, t, t_SMP);
n = find(abs(v_num) == inf); % plot dirac (inf)
stem(t_SMP(n),sign(v_num(n)), 'r*', 'LineWidth', 2);

%plot abs of fourier
subplot(3,1,2); ylabel('abs(X(\omega))'); hold on
ezplot(abs(X_FT), BND_w); hold on; grid on
v_num = subs(abs(X_FT), w, w_SMP);
n = find( abs(v_num) == inf ); % plot dirac (inf)
stem(w_SMP(n),sign(v_num(n)), 'r*', 'LineWidth', 2);

%plot agr of fourier
subplot(3,1,3); ylabel('ARG(X(\omega))'); hold on
ezplot(sym(0), BND_w); hold on; grid on
v_num = subs(angle(X_FT), w, w_SMP);
n = find( abs(v_num) == pi/2 ); % plot dirac (inf)
stem(w_SMP(n),sign(v_num(n)), 'r*', 'LineWidth', 2);
end

function [] =fourT_dirac_o(x, okno, f0)
    y = x * okno;

```

```

    fourT_dirac(y, f0)
end

function [] =fourT_dirac_o_gestosc(x, okno, f0)
    y = x * okno;
    fourT_dirac_gestosc(y, f0)
end

function [] = fourT_cont(x, f0)

syms w t

w0 = 2*pi*f0;
BND_t = [-10/f0;10/f0]; %20 okresow
t_SMP = [BND_t(1):1/(10*f0):BND_t(2)];
BND_w = [-3*w0;3*w0];
w_SMP = [BND_w(1):w0/10:BND_w(2)];
X_FT = fourier(x);

%plot signal
figure
subplot(3,1,1); ylabel('x(t)'); hold on
ezplot(x,BND_t); hold on; grid on;
v_num = subs(x, t, t_SMP);
n = find(abs(v_num) == inf); % plot dirac (inf)
stem(t_SMP(n),sign(v_num(n)),'r*', 'LineWidth', 2);

%plot real of fourier
subplot(3,1,2); ylabel('real(X(\omega))'); hold on
ezplot(real(X_FT), BND_w); hold on; grid on;

%plot imag of fourier
subplot(3,1,3); ylabel('imag(X(\omega))'); hold on
ezplot(imag(X_FT), BND_w); hold on; grid on
end

function [] =fourT_cont_o(x, okno, zakres)
    y = x * okno;
    fourT_cont(y, 10/zakres)
end

function [] =fourT_cont_o_gestosc(x, okno, zakres)
    y = x * okno;
    fourT_cont_gestosc(y, 10/zakres)
end

function [] =fourT_cont_gestosc(x, f0)

syms w t

```

```

w0 = 2*pi*f0;
BND_t = [-10/f0;10/f0]; %20 okresow
t_SMP = [BND_t(1):1/(10*f0):BND_t(2)];
BND_w = [-3*w0;3*w0];
w_SMP = [BND_w(1):w0/10:BND_w(2)];
X_FT = fourier(x);

%plot signal
figure
subplot(3,1,1); ylabel('x(t)'); hold on
ezplot(x,BND_t); hold on; grid on;

%plot abs of fourier
subplot(3,1,2); ylabel('abs(X(\omega))'); hold on
ezplot(abs(X_FT), BND_w); hold on; grid on

%plot arg of fourier
subplot(3,1,3); ylabel('imag(X(\omega))'); hold on
ezplot(angle(X_FT), BND_w); hold on; grid on
end

```