

SPRAWOZDANIE 8

Zajęcia nr 9 - Sygnały cyfrowe

Michał Midor, gr. 5 - 2.12.2025, godz: 15:00

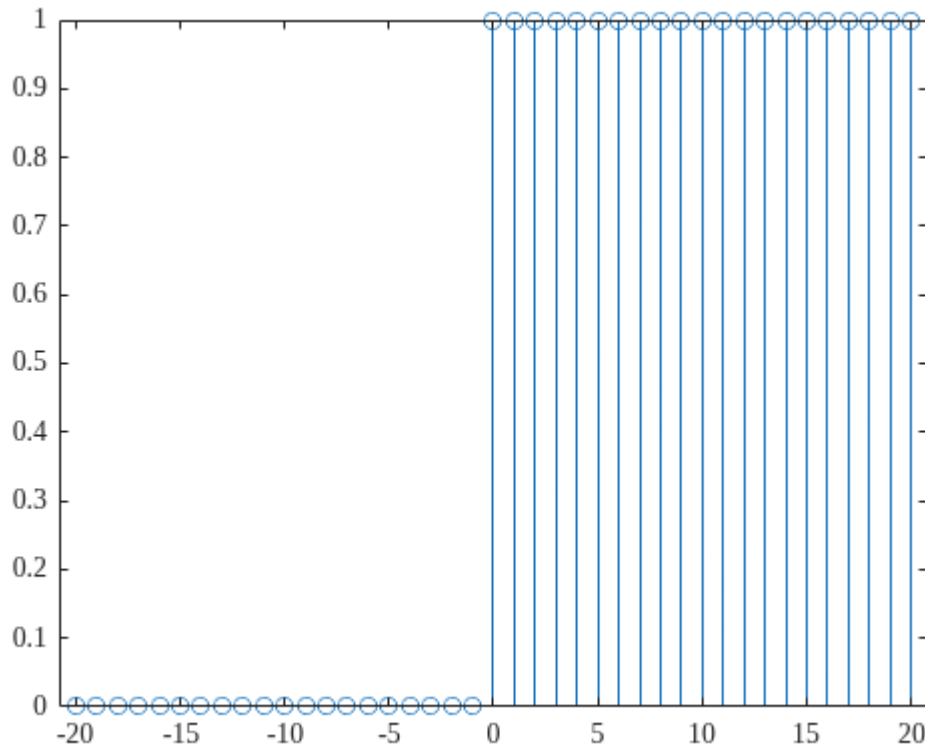
WSTĘP

Sygnał cyfrowy w systemach komputerowych nie jest funkcją ciągłą, lecz **dyskretyzacją** w dwóch domenach: czasu (próbkowanie) i amplitudy (kwantyzacja). W pamięci komputera sygnał to ciąg liczb binarnych. Każda próbka jest reprezentowana przez określoną liczbę bitów, co definiuje rozdzielczość amplitudową. W MATLABIE domyślnym typem danych jest double (podwójna precyzja), a sygnały są przechowywane jako wektory (macierze 1 X N lub N x 1).

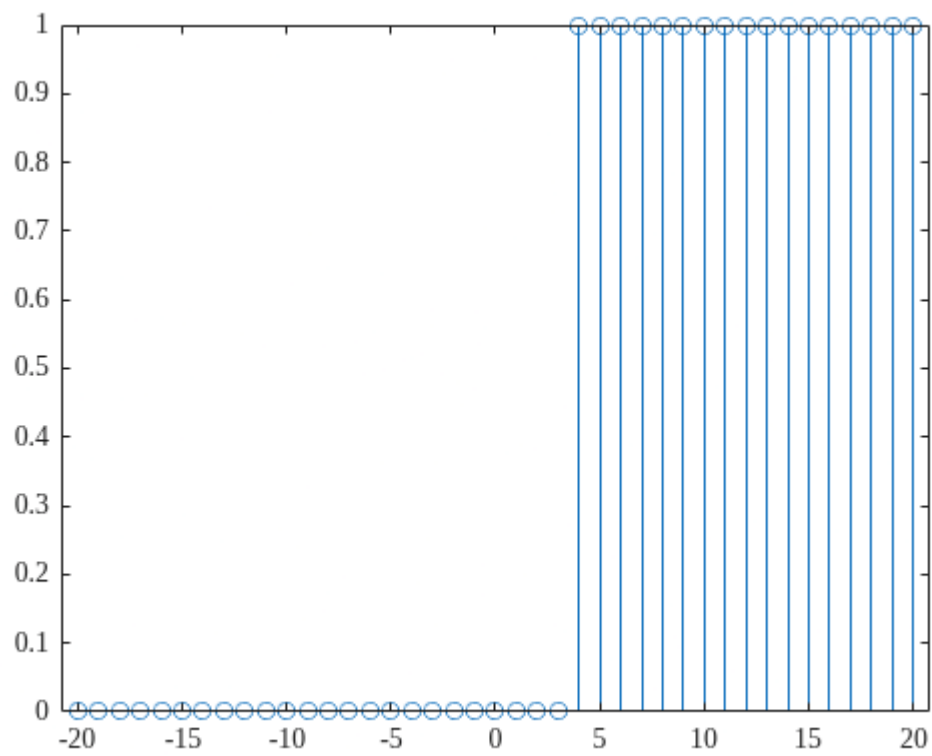
WYKONANIE ĆWICZEŃ

2)

```
t=-20:20;  
u1=zeros(1,41);  
u2=zeros(1,41);  
figure();  
u1(21:end)=ones(1);  
stem(t,u1);
```

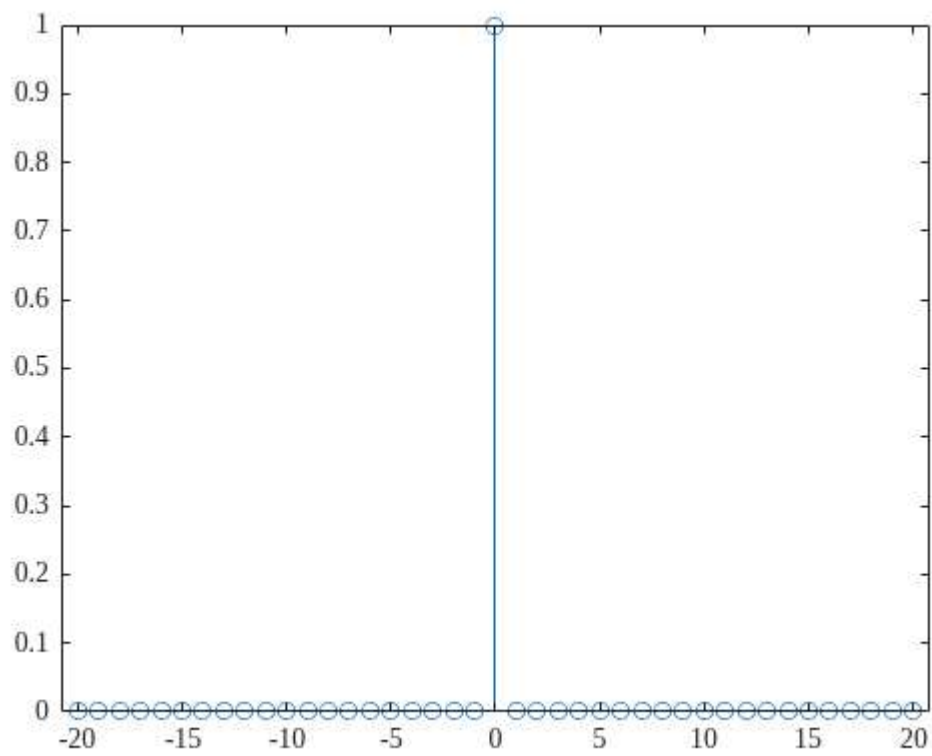


```
u2(25:end)=ones(1);  
stem(t,u2);
```

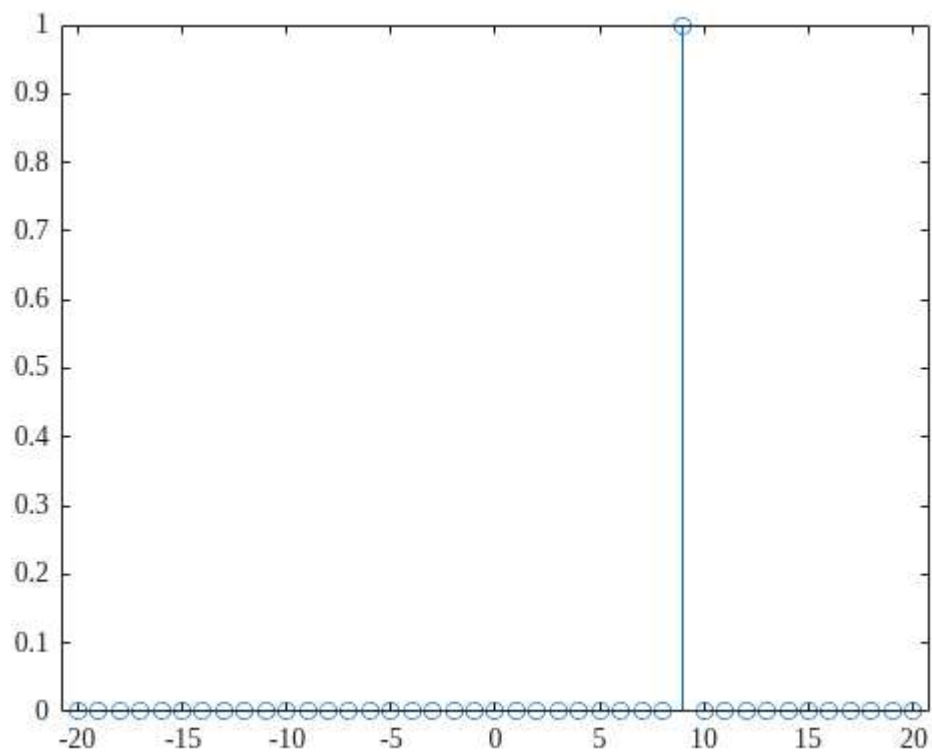


3)

```
figure();  
uk1 = zeros(1,41);  
uk1(21)=ones(1);  
stem(t, uk1);
```

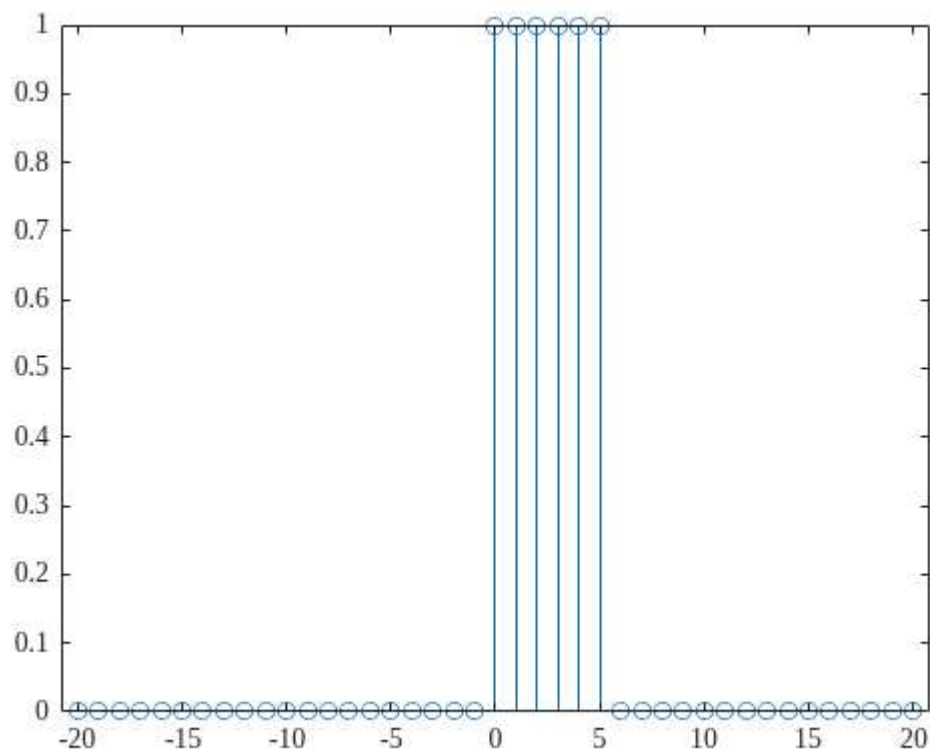


```
uk2 = zeros(1,41);  
uk2(30) = ones(1);  
stem(t, uk2);
```



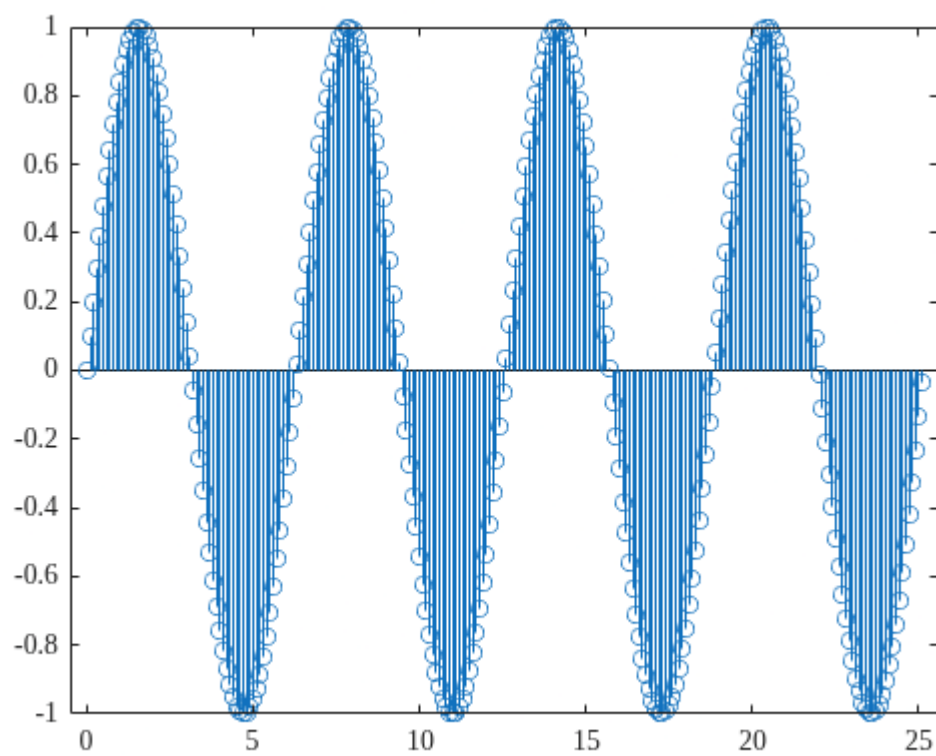
4)

```
u3 = zeros(1,41);  
u3(21:26)=ones(1);  
stem(t,u3);
```

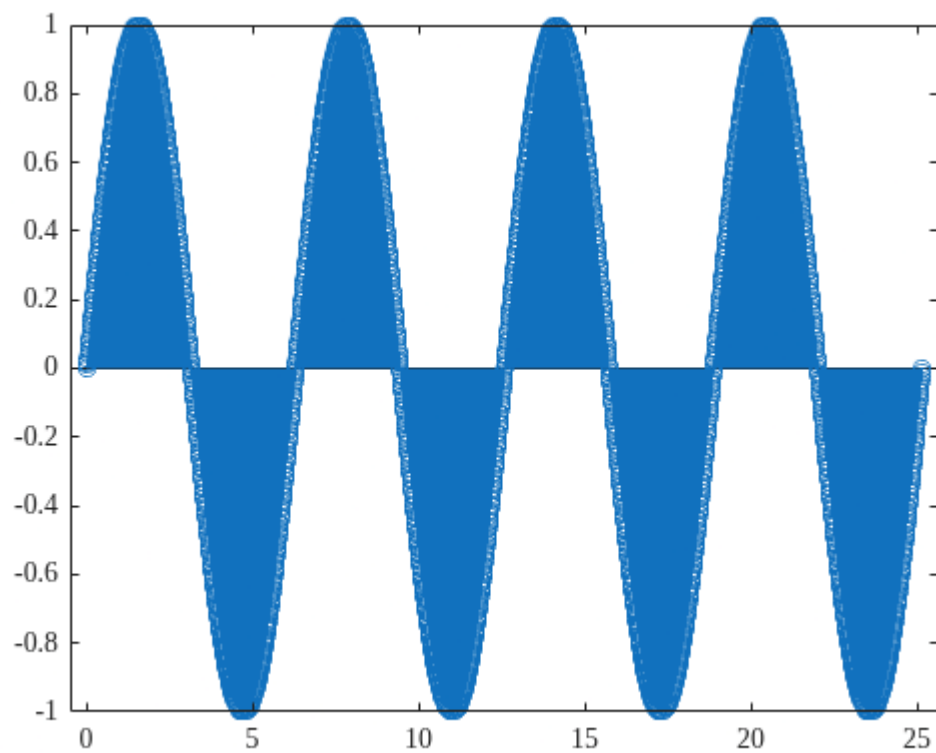


5)

```
t1=0:1/10:8*pi;  
t2=0:1/100:8*pi;  
y_1=sin(t1);  
y_2 = sin(t2);  
figure();  
stem(t1, y_1);
```

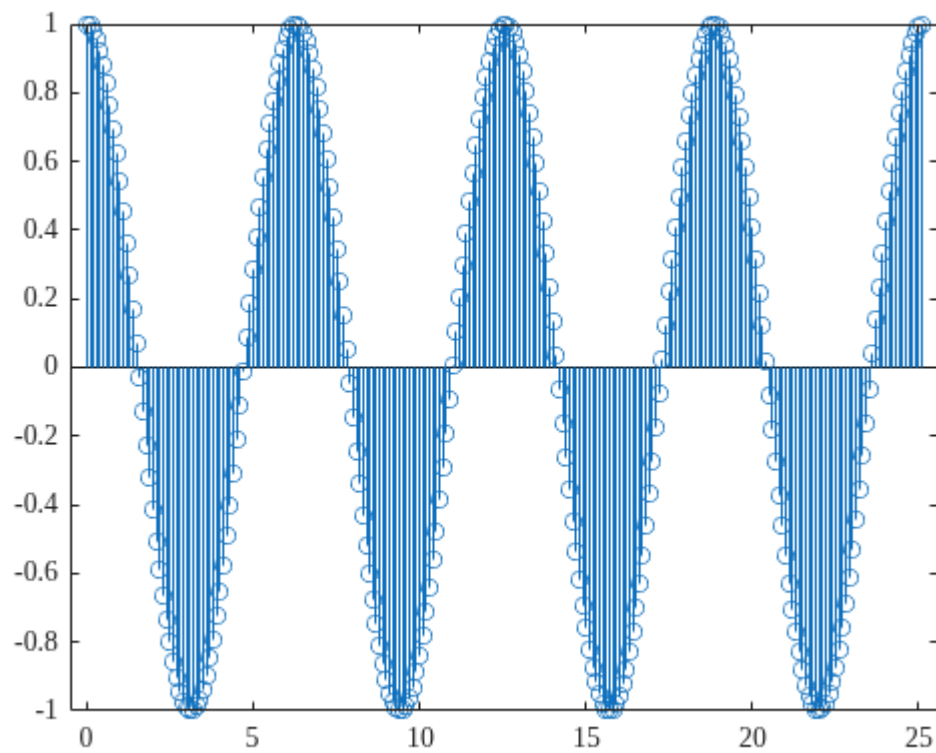


```
stem(t2, y_2);
```

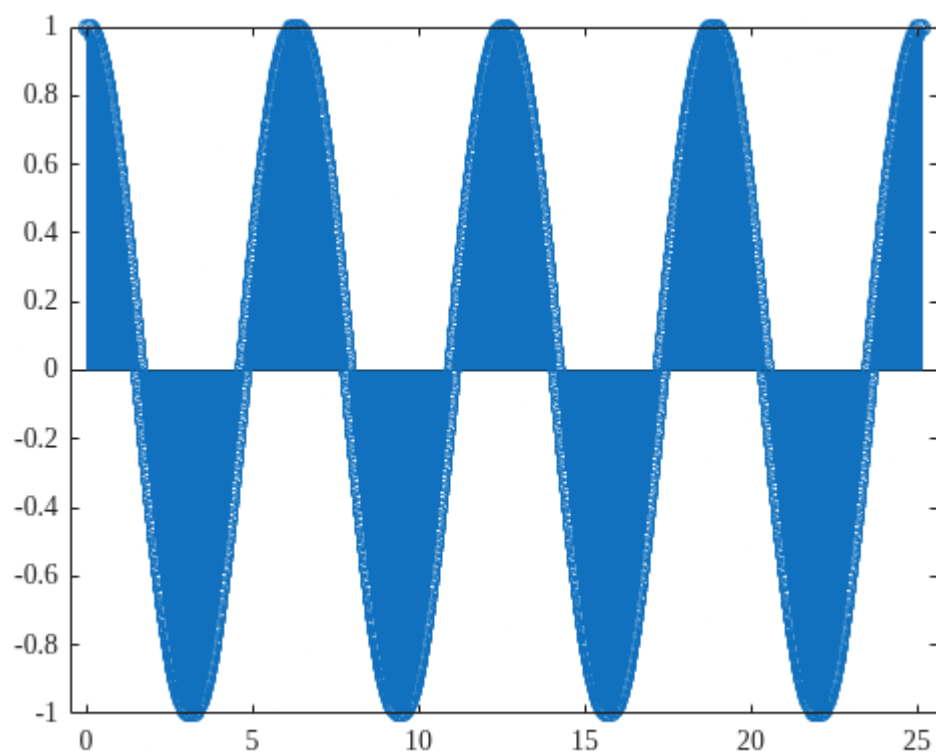


6)

```
y_3 = cos(t1);  
y_4 = cos(t2);  
figure();  
stem(t1, y_3);
```

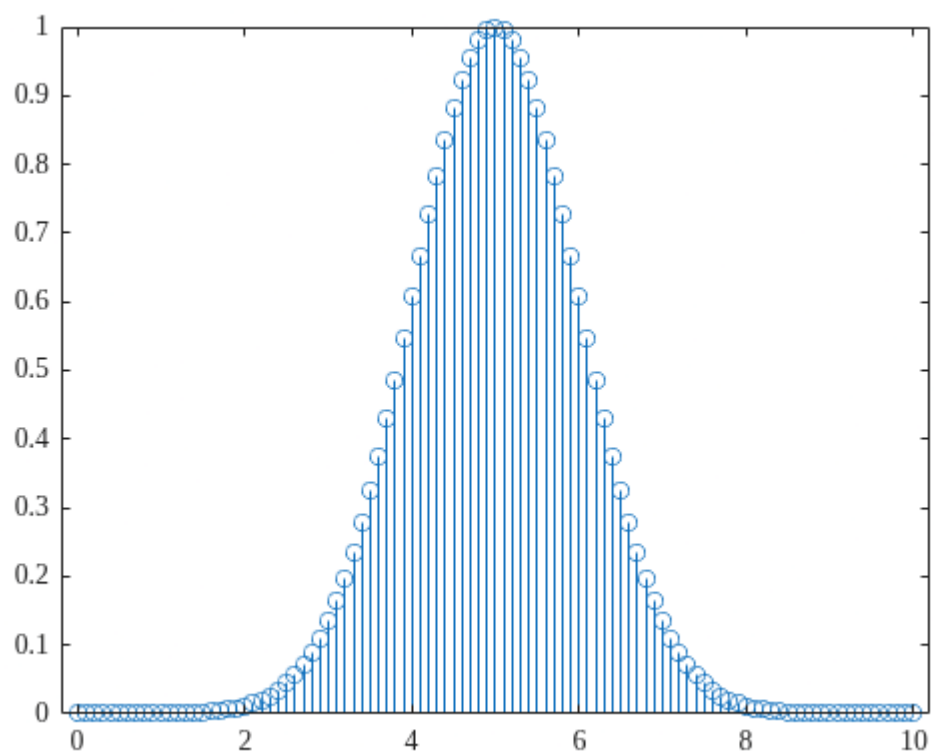


```
stem(t2, y_4);
```



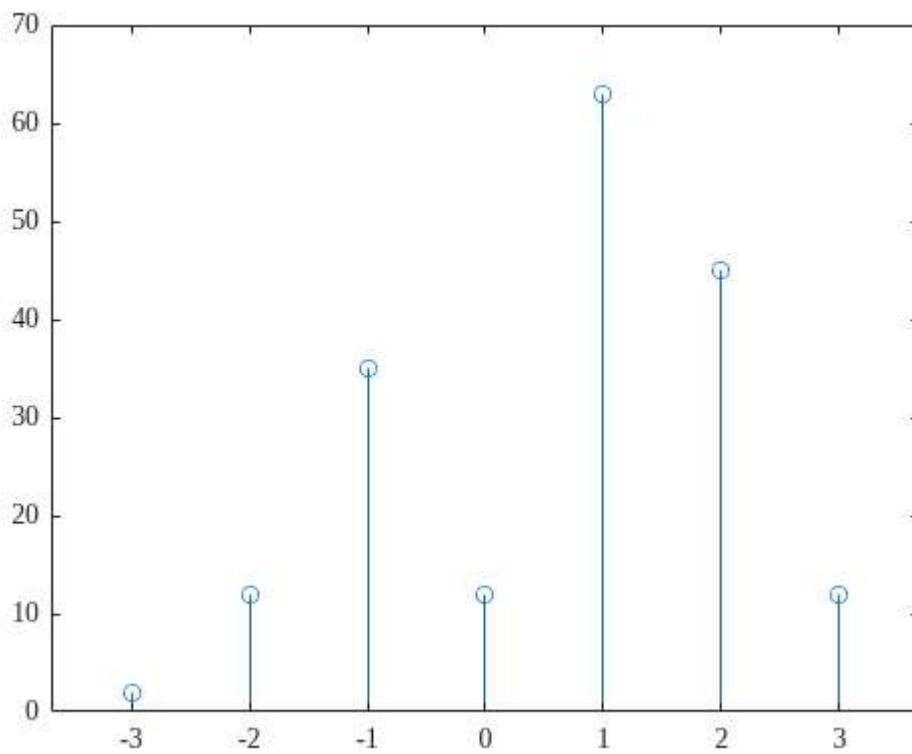
7)

```
t_g = 0:0.1:10;  
y_g = gaussmf(t_g,[1 5]);  
figure();  
stem(t_g, y_g);
```



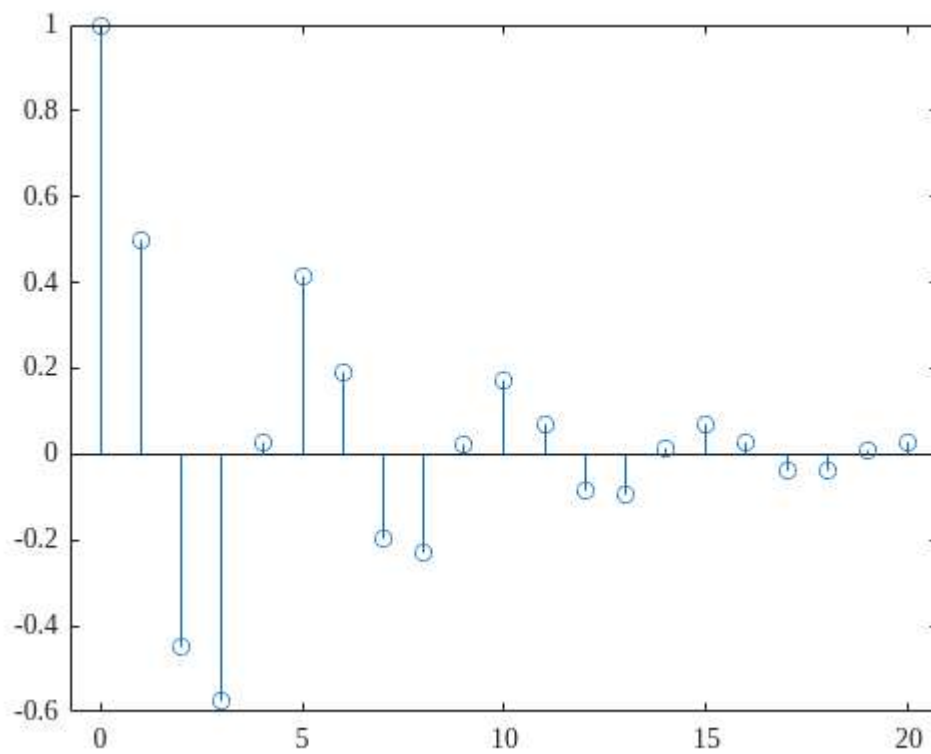
8)

```
y1=[1,3,5,6,7,9,2];  
y2=[2,4,7,2,9,5,6];  
t=[-3 -2 -1 0 1 2 3];  
stem(t, y2.*y1);
```



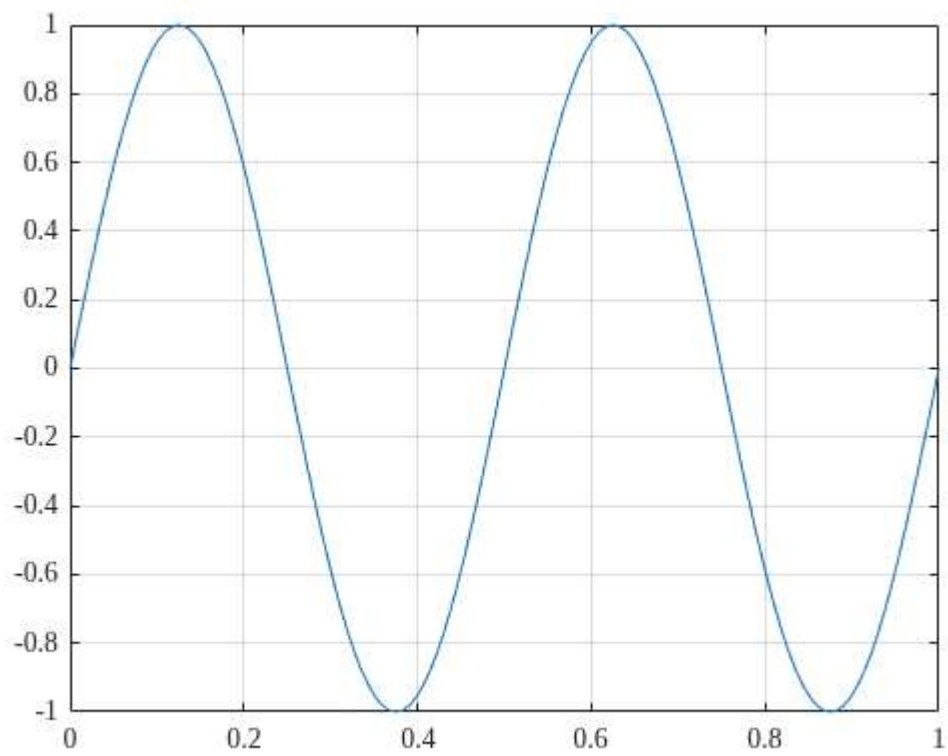
9)

```
n=0:20;  
x=[1,zeros(1,20)];  
a=[1];  
b=[1 -0.5 0.7];  
y=filter(a,b,x);  
stem(n,y);
```



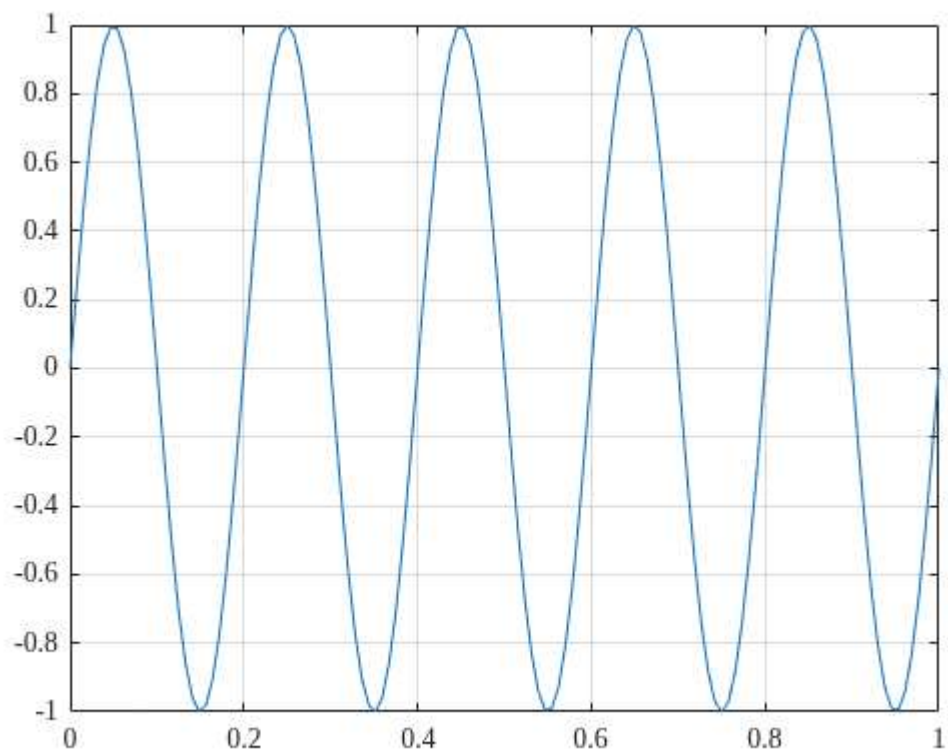
10)

```
n=128;
k=0:n;
t=k/128;
A = 1;
fo = 2;
p = 0;
y = A * sin((2 * pi * fo * t) + p);
plot(t,y);
grid on
```



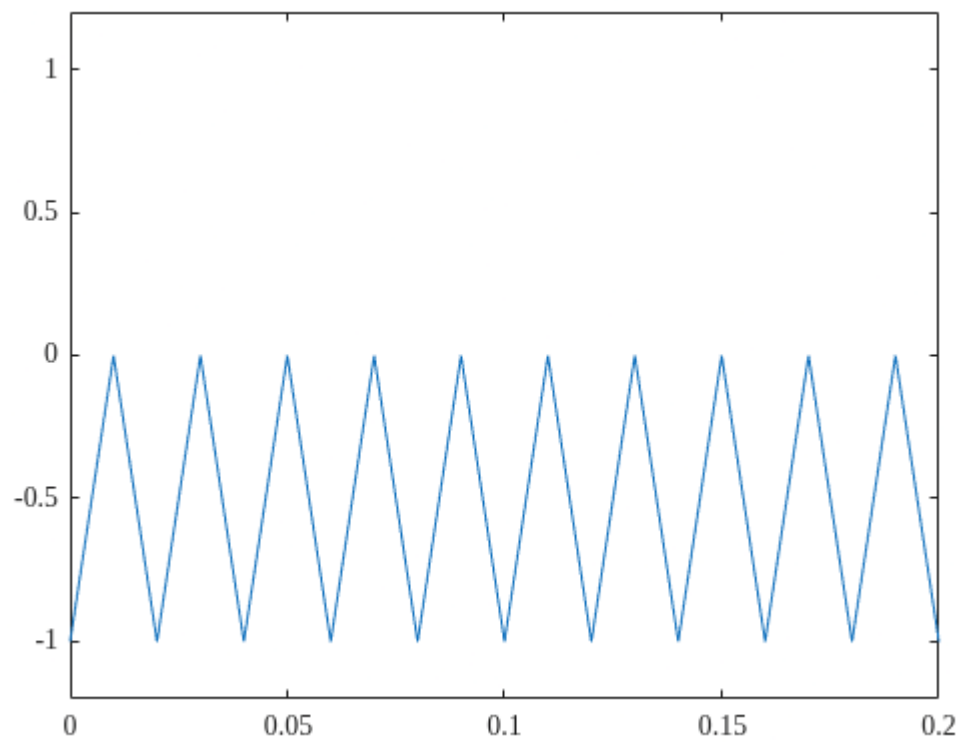
11)

```
f0 = 5;  
y = A * sin((2 * pi * f0 * t) + p);  
plot(t,y);  
grid on
```



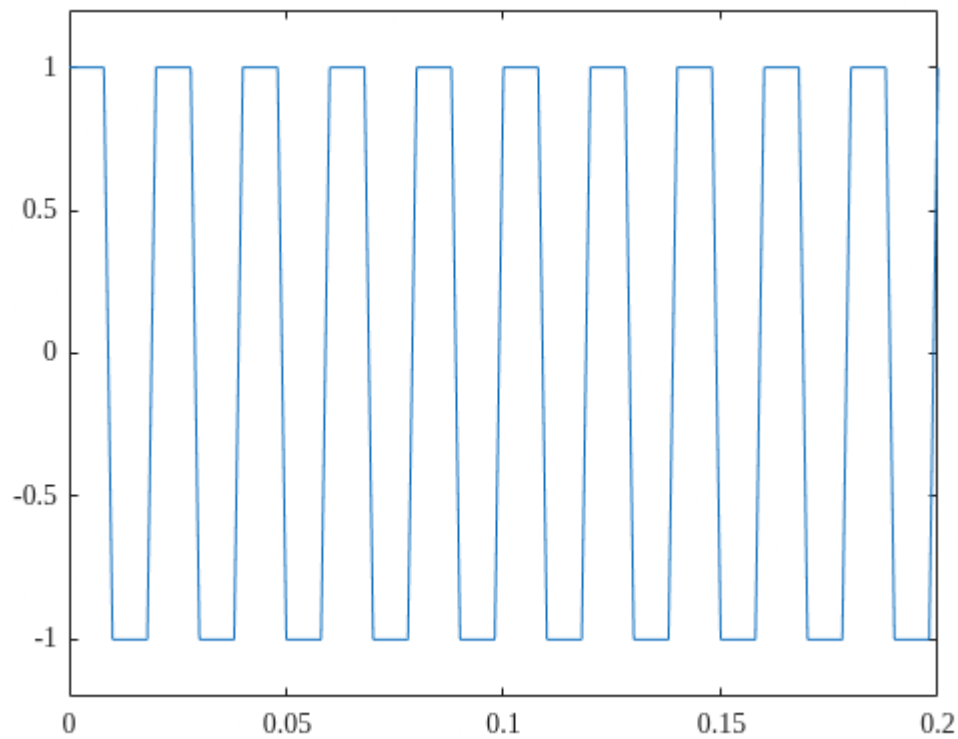
12)

```
fs = 100;  
t = 0:1/fs:1.5;  
x1 = sawtooth(2*pi*50*t);  
plot(t,x1)  
axis([0 0.2 -1.2 1.2]);
```



13)

```
fs_sq=500;  
t = 0:1/fs_sq:1.5;  
x2 = square(2*pi*50*t);  
plot(t,x2)  
axis([0 0.2 -1.2 1.2])
```



14)

```
fs = 10000;
t = -1:1/fs:1;
x1 = tripuls(t,80e-3);
x2 = rectpuls(t,80e-3);
subplot(2,1,1)
plot(t,x1)
axis([-0.1 0.1 -0.2 1.2])
xlabel('Time (sec)')
ylabel('Amplitude')
subplot(2,1,2)
plot(t,x2)
axis([-0.1 0.1 -0.2 1.2])
xlabel('Time (sec)')
ylabel('Amplitude')
```

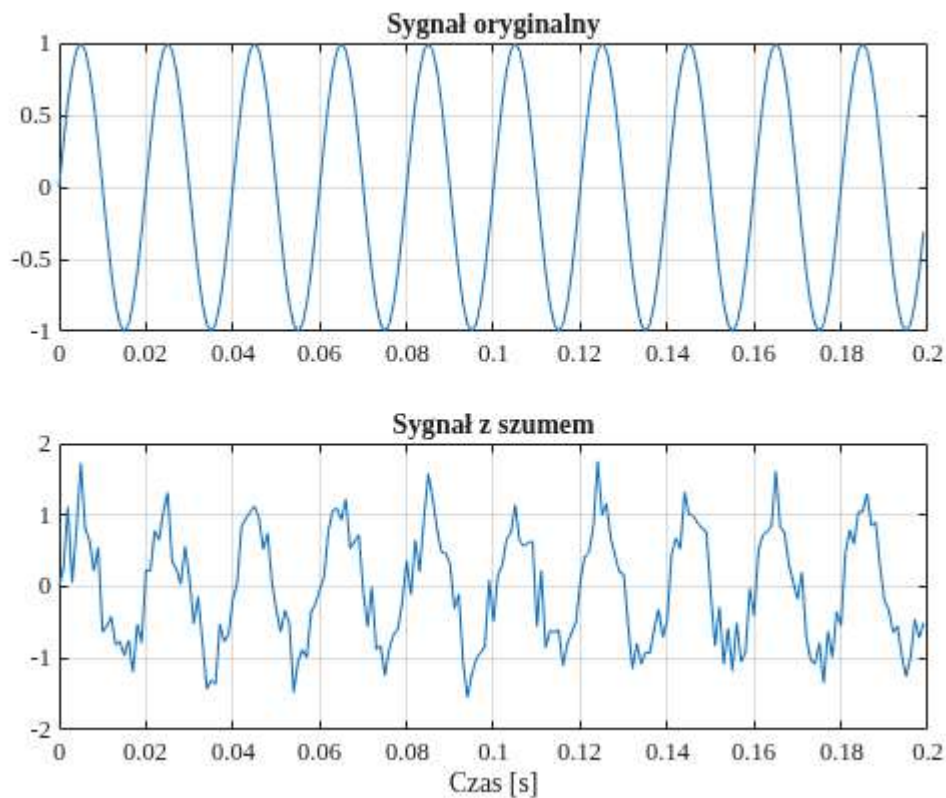
15)

```
T = 10*(1/50);
fs = 1000;
t = 0:1/fs:T-1/fs;
% Sygnał sinusoidalny z szumem
x_c = sin(2*pi*50*t);
szum = 0.3*randn(size(t));
% szum gaussowski
```

```

x_szum = x_c + szum;
subplot(2,1,1)
plot(t,x_c)
title('Sygnał oryginalny')
grid on
subplot(2,1,2)
plot(t,x_szum)
title('Sygnał z szumem')
grid on
xlabel('Czas [s]')

```



16)

```

T_mod = 0.2;
fs = 1000;
t_mod = 0:1/fs:T_mod-1/fs;
fn = 100;%częst. nośna [Hz]
fm = 80;%częst. modulująca [Hz]
An = 5; % amplituda nośna
Am = 0.3; % amplituda modulująca

x_original = An*sin(2*pi*fn*t_mod);
x_mod = Am*sin(2*pi*fm*t_mod);

x_final = (1 + x_mod).*x_original;

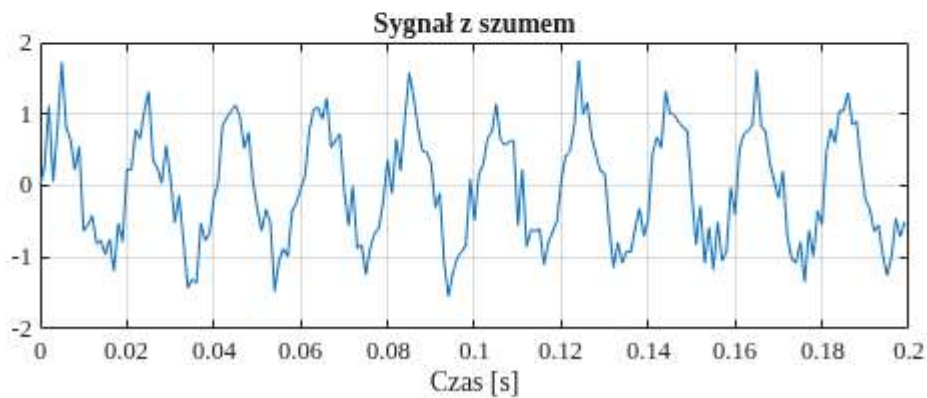
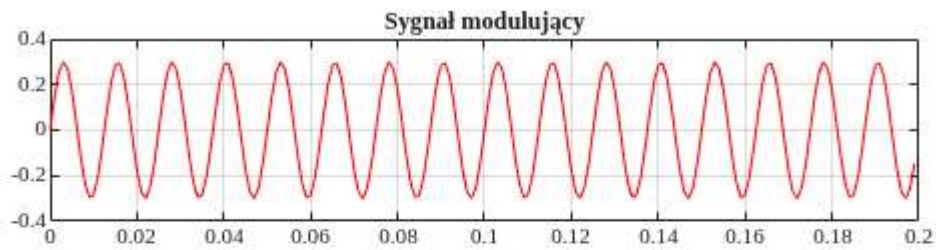
subplot(3,1,1)

```

```

plot(t_mod, x_mod, 'r')
title('Sygnał modulujący')
grid on

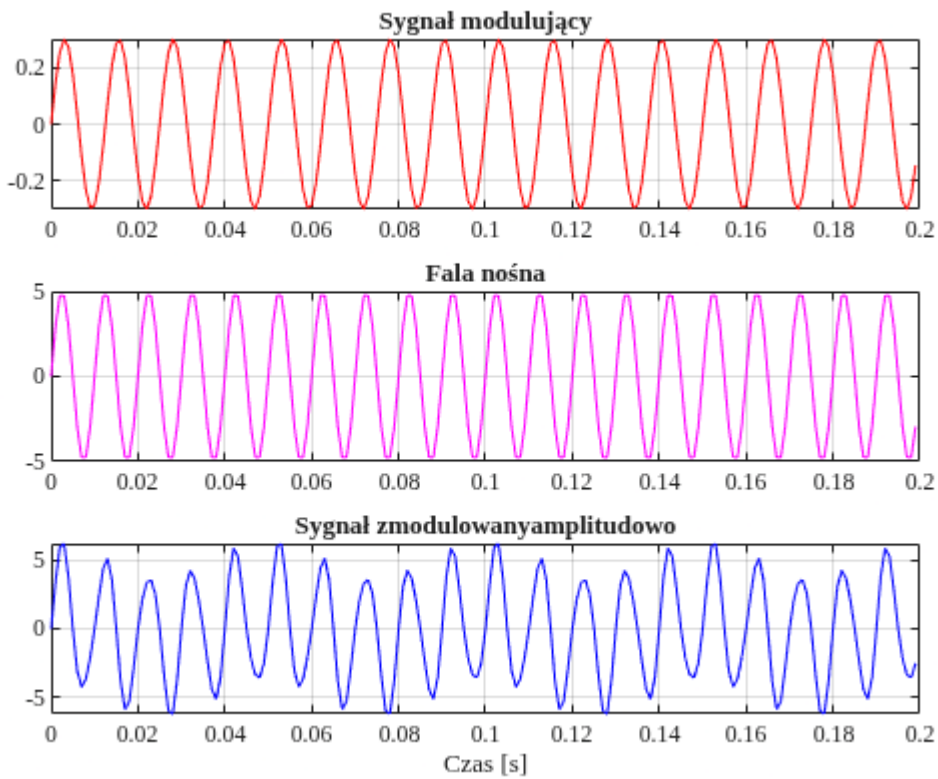
```



```

subplot(3,1,2)
plot(t_mod, x_original, 'm')
title('Fala nośna')
grid on
subplot(3,1,3)
plot(t_mod, x_final, 'b')
title('Sygnał zmodulowanyamplitudowo')
grid on
xlabel('Czas [s]')

```

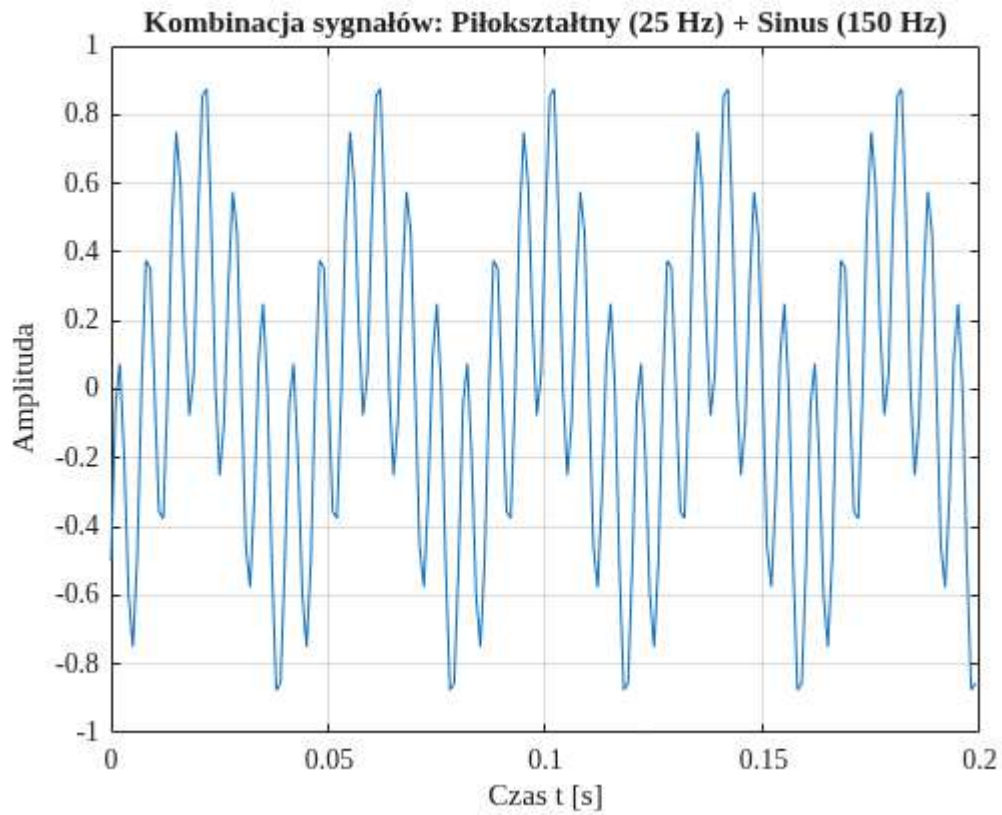


17)

```
T_mix = 10*(1/50);
fs_mix = 1000;
t_mix = 0:1/fs:T_mix-1/fs;

x1 = 0.5*sawtooth(2*pi*25*t_mix, 0.5);
x2 = 0.5*sin(2*pi*150*t_mix);

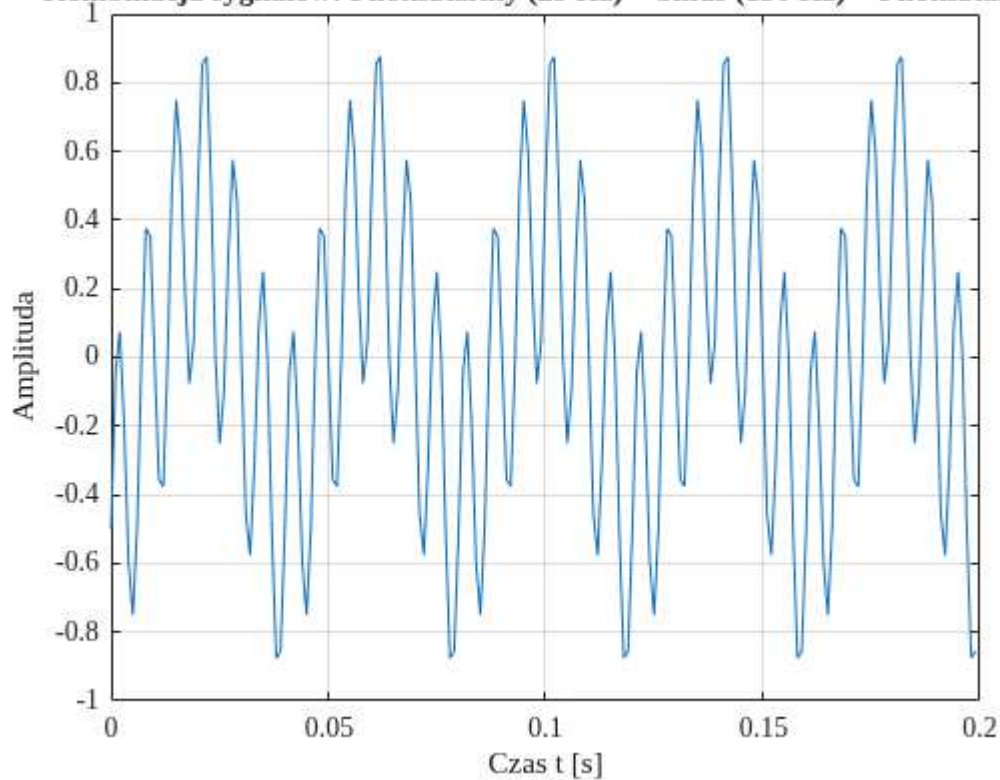
x_mix = x1+x2;
figure();
plot(t, x_mix);
title('Kombinacja sygnałów: Piłokształtny (25 Hz) + Sinus (150 Hz)');
xlabel('Czas t [s]');
ylabel('Amplituda');
grid on;
```



18)

```
x3 = 0.5*tripuls(t_mix, 0.500);
x_mix_2 = x1+x2+x3;
figure();
plot(t, x_mix);
title('Kombinacja sygnałów: Piłokształtny (25 Hz) + Sinus (150 Hz) +  
Piłokształtny');
xlabel('Czas t [s]');
ylabel('Amplituda');
grid on;
```

Kombinacja sygnałów: Piłokształtny (25 Hz) + Sinus (150 Hz) + Piłokształtny



19)

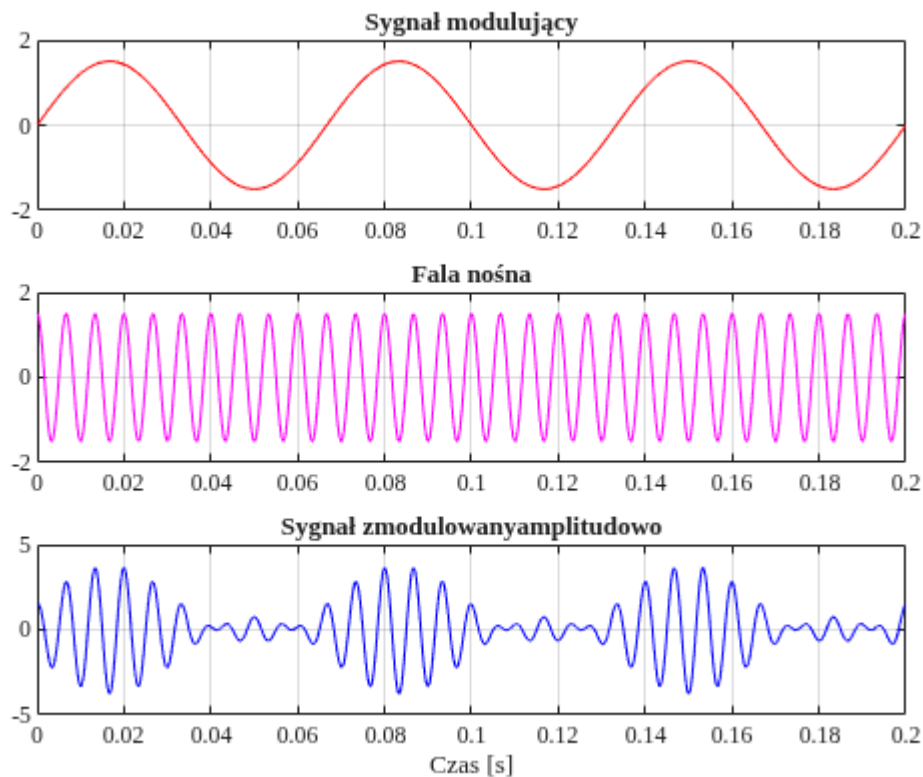
```
T = 0.2; % czas trwania [s]
fs = 10000; % częstotliwość próbkowania [Hz]
t = 0:1/fs:T-1/fs;
fn = 150; % częstotliwość nośna [Hz]
fm = 15; % częstotliwość modulująca [Hz]
Am = 1.5; % amplituda sygnału modulującego
An = 1.5; % amplituda sygnału nośnego
im = 8; % indeks modulacji

nosny = An * cos(2*pi*fn*t);
modulujacy = Am * sin(2*pi*fm*t);

zmodulowany = (1+modulujacy).*nosny;

subplot(3,1,1)
plot(t, modulujacy, 'r')
title('Sygnał modulujący')
grid on
subplot(3,1,2)
plot(t, nosny, 'm')
title('Fala nośna')
grid on
subplot(3,1,3)
plot(t, zmodulowany, 'b')
```

```
title('Sygnał zmodulowanyamplitudowo')
grid on
xlabel('Czas [s]')
```



ODPOWIEDZI NA PYTANIA

Jakie informacje może zawierać sygnał cyfrowy?

Sygnał może zawierać:

- **Dane multimedialne:** Dźwięk, obraz, wideo.
- **Dane telemetryczne/sterujące:** Odczyty z czujników (temperatura, ciśnienie), sygnały PWM do sterowania silnikami

Czy da się zamienić sygnały cyfrowe na sygnały analogowe?

Proces ten realizowany jest przez przetwornik C/A (DAC). Polega on na splotowaniu ciągu impulsów (próbek) z funkcją interpolującą. Fizycznie, na wyjściu DAC stosuje się **analogowy filtr rekonstrukcyjny**, który wygładza sygnał, usuwając obrazy widmowe powstałe w procesie próbkowania.

Na czym polega różnica pomiędzy przekazem analogowym a cyfrowym w telewizji?

- **Analogowa:** Obraz i dźwięk modulują bezpośrednio falę nośną (AM dla wizji, FM dla fonii). Jakość sygnału spada liniowo wraz ze wzrostem szumu (śnieżenie obrazu).
- **Cyfrowa:** Przesyłany jest strumień bitów (często skompresowany, np. MPEG). Stosuje się modulacje cyfrowe.

Czy sygnał cyfrowy może mieć wiele wymiarów?

Tak, a każdy z nich odpowiada osobnemu typowi danych niesionych przez sygnał, przykładowo:

- **1D (Wektor):** Sygnał dźwiękowy, odczyt temperatury w czasie.
- **2D (Macierz):** Obraz statyczny (intensywność piksela).
- **3D :** Wideo (obraz 2D w czasie)
- **Wielowymiarowość (N-D):** Dane z sieci sensorów, gdzie każda zmienna to osobny wymiar w przestrzeni cech.

Czy częstotliwość próbkowania jest ważna?

Bardzo ważna, jej wartość wpływa na dokładność rekonstrukcji sygnału spróbkowanego.

Po co stosuje się modulację?

Modulacja to proces translacji widma sygnału użytecznego w obszar wysokich częstotliwości. Technicznie realizuje się to poprzez zmianę parametrów fali nośnej (amplitudy, częstotliwości lub fazy).

Główne inżynierskie powody stosowania modulacji:

1. **Dopasowanie do kanału:** Anteny są skuteczne, gdy ich długość jest porównywalna z długością. Dla mowy ludzkiej (niskie częstotliwości) antena musiałaby mieć kilometry długości.
2. **Multipleksacja:** Umożliwienie przesyłu wielu sygnałów w tym samym medium (powietrzu, kablu) na różnych częstotliwościach.