2. SEMNALE ŞI SISTEME ÎN TIMP DISCRET

2.1. Semnale în timp discret

Un semnal în timp discret este definit prin valorile acestuia măsurate la momente discrete de timp. Semnalele în timp discret sunt reprezentate matematic prin secvențe de numere notate:

$$x[n] \qquad , \qquad N_1 \le n \le N_2 \tag{2.1}$$

În MATLAB aceste secvențe se pot defini ca vectori linie sau coloană, având elemente reale sau complexe. O primă limitare apare din faptul că aceşti vectori sunt de lungime finită în timp ce în problemele de prelucrarea numerică a semnalelor se poate lucra cu secvențe de lungime infinită.

2.1.1. Definirea semnalelor în timp discret

În studiul semnalelor şi sistemelor în timp discret se utilizează câteva secvențe de bază ce vor fi prezentate în continuare, împreună cu modul lor de definire în MATLAB.

- ***Impulsul unitate***

Din punct de vedere matematic este definit astfel:

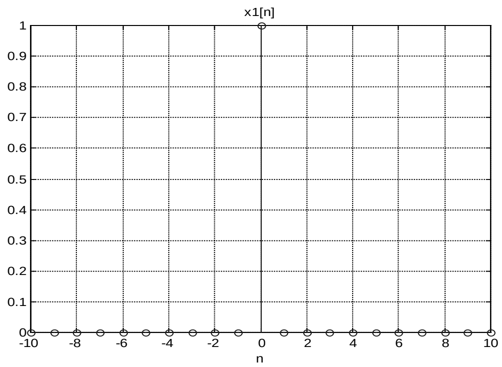
$$\delta[n] = \begin{cases} 1 & , \quad n = 0 \\ 0 & , \quad n \ne 0 \end{cases} \tag{2.2}$$

Utilizând proprietatea de deplasare în timp se poate scrie că

$$\delta[n-n_0] = \begin{cases} 1 & , \quad n = n_0 \\ 0 & , \quad n \ne n_0 \end{cases} \tag{2.3}$$

Deoarece în MATLAB nu putem defini secvențe de lungime infinită trebuie precizat domeniul de valori pentru *n*.

***Exemple*:**
Să se definească şi să se reprezinte grafic secvențele:
1. $x_1[n]=\delta[n]$
2. $x_2[n]=\delta[n-1]$
3. $x_3[n]=\delta[n+1]$
4. $x_4[n]=0{,}5\delta[n-3]$, pentru $-10\le n\le 10$.

Toți vectorii vor avea deci 21 de elemente. Se poate proceda în mai multe moduri:
```
n=-10:10;
x1=[zeros(1,10),1,zeros(1,10)];
stem(n,x1),grid,title('x1[n]'),xlabel('n')
```


Ținând cont însă de faptul că secvența are un singur eşantion nenul se poate folosi o altă exprimare mai facilă:
```
x1=zeros(size(n));
x1(11)=1;
```
// deoarece $n=-10,-9,\ldots,-1,0,1,\ldots,9,10$, momentul de timp $n=0$ corespunde celui de al 11-lea element al vectorului, a cărui nouă valoare va fi 1.
În mod asemănător se pot defini şi reprezenta grafic celelalte secvențe:
```
x2=zeros(size(n));
x2(12)=1;
stem(n,x2),grid,title('x2[n]'),xlabel('n')
```
// momentul de timp $n=1$ corespunde celui de al 12-lea element al vectorului.
```
x3=zeros(size(n));
x3(10)=1;
stem(n,x3),grid,title('x3[n]'),xlabel('n')
```
// momentul de timp $n=-1$ corespunde celui de al 10-lea element al vectorului.
```
x4=zeros(size(n));
x4(14)=0.5;
stem(n,x4),grid,title('x4[n]'),xlabel('n')
```
// momentul de timp $n=3$ corespunde celui de al 14-lea element al vectorului.

E1. ***Exerciții*:**

Să se definească şi să se reprezinte grafic următoarele secvențe:

1. $x_1[n] = 0{,}9\delta[n-5]$ pentru $1 \le n \le 20$
2. $x_2[n] = 0{,}8\delta[n]$ pentru $-15 \le n \le 15$
3. $x_3[n] = 1{,}5\delta[n-333]$ pentru $300 \le n \le 350$
4. $x_4[n] = 4{,}5\delta[n+7]$ pentru $-10 \le n \le 10$
5. $x_5[n] = \delta[n] - \delta[n-1]$ pentru $-10 \le n \le 10$
6. $x_6[n] = \delta[n] - 0{,}5\delta[n-1] + 0{,}3\delta[n-2] - 2\delta[n+1]$ pentru $-10 \le n \le 10$

- ***Treapta unitate***

Din punct de vedere matematic este definit astfel:

$$u[n] = \begin{cases} 1 & , \quad n \ge 0 \\ 0 & , \quad n < 0 \end{cases} \tag{2.4}$$

Utilizând proprietatea de deplasare în timp se poate scrie că

$$u[n-n_0] = \begin{cases} 1 & , \quad n \ge n_0 \\ 0 & , \quad n < n_0 \end{cases} \tag{2.5}$$

Reamintim că în MATLAB nu putem defini secvențe de lungime infinită şi din această cauză trebuie precizat domeniul de valori pentru *n*.

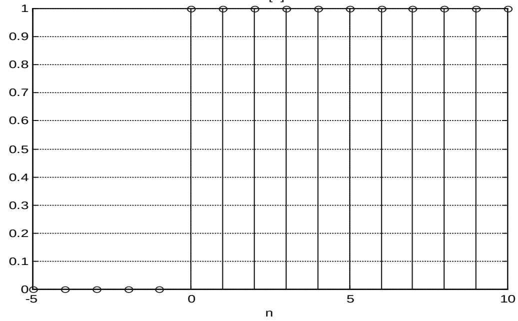
***Exemple*:**

Să se definească şi să se reprezinte grafic secvențele:

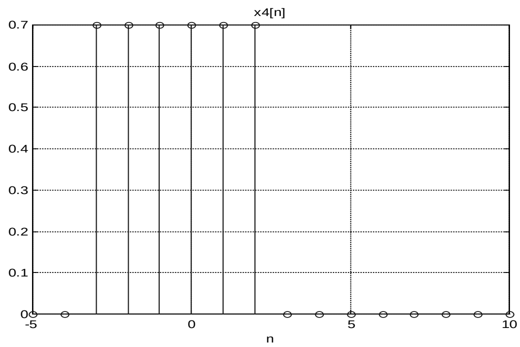
1. $x_1[n] = u[n]$
2. $x_2[n] = u[n-2]$
3. $x_3[n] = u[n+2]$
4. $x_4[n] = 0{,}7(u[n+3] - u[n-3])$, pentru $-5 \le n \le 10$.

Toți vectorii vor avea deci 16 elemente. Momentul de timp $n = 0$ corespunde celui de al 6-lea element al vectorului. Secvențele se vor obține prin concatenarea unor vectori cu elemente nule şi a unor vectori cu elemente diferite de zero, în funcție de momentul de timp în care apare primul eşantion nenul.

```
n=-5:10;
x1=[zeros(1,5),ones(1,11)];
stem(n,x1),grid,title('x1[n]'),xlabel('n')
```



```
x2=[zeros(1,7),ones(1,9)];
stem(n,x2),grid,title('x2[n]'),xlabel('n')
```
// momentul de timp $n=2$ corespunde celui de al 8-lea element al vectorului.
```
x3=[zeros(1,3),ones(1,13)];
stem(n,x3),grid,title('x1[n]'),xlabel('n')
```
// momentul de timp $n=-2$ corespunde celui de al 4-lea element al vectorului.
```
x4=0.7*[zeros(1,2),ones(1,6),zeros(1,8)];
stem(n,x4),grid,title('x4[n]'),xlabel('n')
```



// secvenţa $u[n+3]$ are primul eşantion nenul la momentul de timp $n=-3$ (elementul al 3-lea din vector); secvenţa $u[n-3]$ are primul eşantion nenul la momentul de timp $n=3$ (elementul al 9-lea din vector) astfel încât începând din acest moment secvenţa $x_4[n]$ are toate eşantioanele egale cu zero.

E2. ***Exerciţii*:**

Să se definească şi să se reprezinte grafic următoarele secvenţe:

1. $x_1[n]=0{,}3u[n]$ pentru $-10\le n\le 20$
2. $x_2[n]=u[n-7]$ pentru $0\le n\le 30$
3. $x_3[n]=1{,}8u[n+3]$ pentru $-15\le n\le 15$
4. $x_4[n]=u[n]-u[n-6]$ pentru $-10\le n\le 20$
5. $x_5[n]=u[n]+0{,}5u[n-4]-0{,}5u[n+4]$ pentru $-10\le n\le 10$
6. $x_6[n]=\delta[n-1]+u[n-5]+\delta[n+2]-2\delta[n-9]$ pentru $-10\le n\le 20$

- ***Semnale periodice***

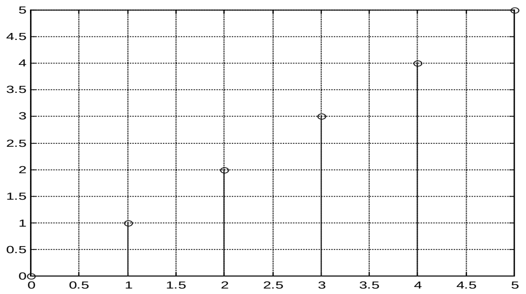
În MATLAB nu se pot genera secvenţe de lungime infinită astfel încât trebuie precizat numărul de perioade pentru o anumită secvenţă.

***Exemple*:**

Să se definească şi să se reprezinte grafic secvenţele “periodice”:

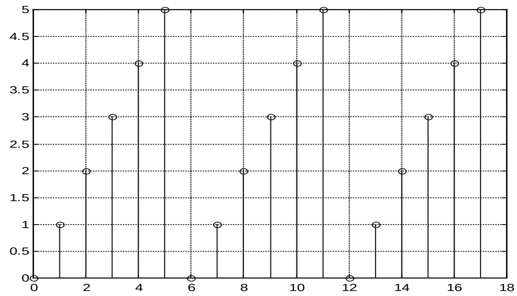
1. $x_1[n]=n$ pentru $0\le n\le 5$ (3 perioade)
2. $x_2[n]=u[n-2]-u[n-4]$ pentru $0\le n\le 5$ (5 perioade)
3. $x_3[n]=\delta[n-1]-\delta[n-3]$ pentru $0\le n\le 5$ (7 perioade)

```
n=0:5;
x1=n;
stem(n,x1),grid
```



// s-a generat o singură perioadă.

```
x11=[x1,x1,x1];
stem(0:(length(x11)-1),x11),grid
```



// s-au generat cele 3 perioade ale secvenţei $x_1[n]$; *explicaţi* de ce s-a folosit sintaxa respectivă pentru `stem`.

Dacă însă numărul de perioade este mai mare (de exemplu 10) modalitatea de definire de mai sus ar presupune scrierea unui vector de tipul x11=[x1,x1,…,x1] (de 10 de ori x1), ceea ce ar fi dificil din punct de vedere al operatorului. În acest caz se preferă un mod de definire mai facil:

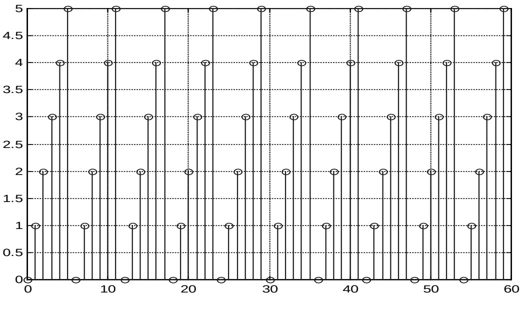
```
x11=x1’*ones(1,10);
```

// se obţine o matrice cu 10 de coloane, pe fiecare coloană aflându-se elementele vectorului x1 (s-a folosit transpunerea deoarece x1 era definit ca vector linie).

```
x11=x11(:);
```

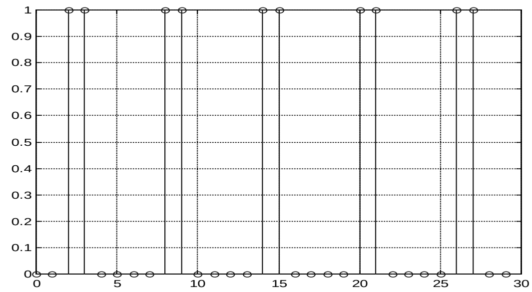
// matricea respectivă devine un vector coloană (vezi secţiunea 1.2.3. din capitolul 1).

```
stem(0:(length(x11)-1),x11),grid
```

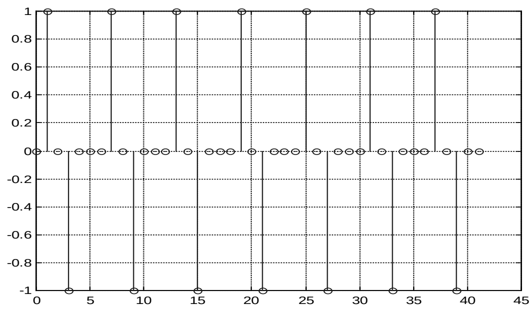


În mod asemănător se pot defini şi reprezenta grafic celelalte secvenţe:

```
x2=[0;0;1;1;0;0];
x22=x2*ones(1,5);
x22=x22(:);
stem(0:(length(x22)-1),x22),grid
```



```
x3=[0;1;0;-1;0;0];
x33=x3*ones(1,7);
x33=x33(:);
stem(0:(length(x33)-1),x33),grid
```



E3. ***Exerciţii*:**

Să se definească şi să se reprezinte grafic următoarele secvenţe:

1. $x_1[n]=u[n]-u[n-4]$ pentru $0 \le n \le 10$ (8 perioade)
2. $x_2[n]=\delta[n]-\delta[n-1]+\delta[n-2]-\delta[n-3]$ pentru $0 \le n \le 5$ (6 perioade)
3. $x_3[n]=0{,}5+0{,}5\delta[n-1]-u[n-3]+u[n-6]$ pentru $0 \le n \le 8$ (4 perioade)

- ***Alte tipuri de semnale***

Folosind operatorii şi funcţiile matematice prezentate în secţiunile 1.2.9. şi 1.2.10. din capitolul 1 se pot defini şi alte tipuri de secvenţe numerice.

***Exemple*:**

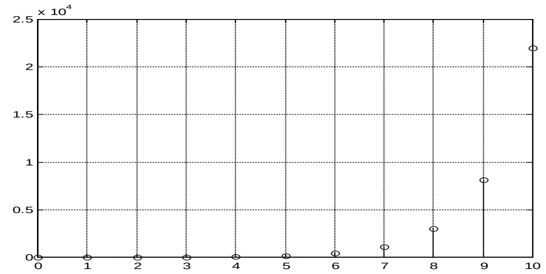
Să se definească şi să se reprezinte grafic secvenţele:

1. $x_1[n]=e^n$
2. $x_2[n]=(-1)^n\cos\left(\dfrac{n\pi}{15}\right)$

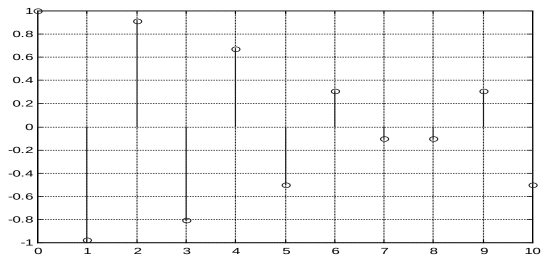
3. $x_3[n] = \sqrt{\left|n - \frac{1}{2}\right|}$

4. $x_4[n] = 1 - \log_{10}(3n+2)$, pentru $0 \le n \le 10$

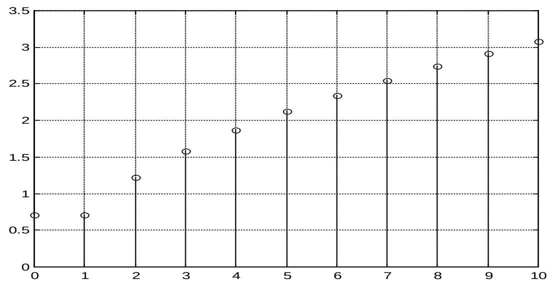
```
n=0:10;
x1=exp(n);
stem(n,x1),grid
```



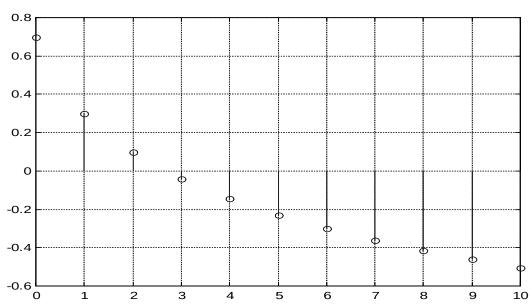

```
x2=((-1).^n).*cos(n*pi/15);
stem(n,x2),grid
```




```
x3=sqrt(abs(n-1/2));
stem(n,x3),grid
```




```
x4=1-log10(3*n+2);
stem(n,x4),grid
```



E4. ***Exerciţii*:**

Să se definească şi să se reprezinte grafic următoarele secvenţe:

1. $x_1[n]=(-1)^n$ pentru $0\le n\le 10$
2. $x_2[n]=\left(\frac{1}{2}\right)^n-\left(-\frac{1}{2}\right)^n$ pentru $0\le n\le 10$
3. $x_3[n]=2\sin\left(\frac{n\pi}{11}\right)e^{n-1}$ pentru $-20\le n\le 20$
4. $x_4[n]=\ln\left|\cos\left(\frac{n\pi}{15}\right)-\sin\left(\frac{n\pi}{15}\right)\right|$ pentru $-20\le n\le 20$
5. $x_5[n]=\left(1+(-1)^n\right)\sin\left(3\pi n+\frac{\pi}{2}\right)$ pentru $-20\le n\le 20$
6. $x_6[n]=x_5[n]\left(u[n]-u[n-10]\right)$
7. $x_7[n]=x_3[n]\left(\delta[n]+0{,}5\delta[n-10]\right)$
8. $x_8[n]=x_4[n]u[n-5]$
9. $x_9[n]=\begin{cases}-2 & , \; -22\le n\le -8\\ 2 & , \; -7\le n\le 7\\ -3 & , \; 8\le n\le 22\\ 0 & , \; 23\le n\le 32\end{cases}$

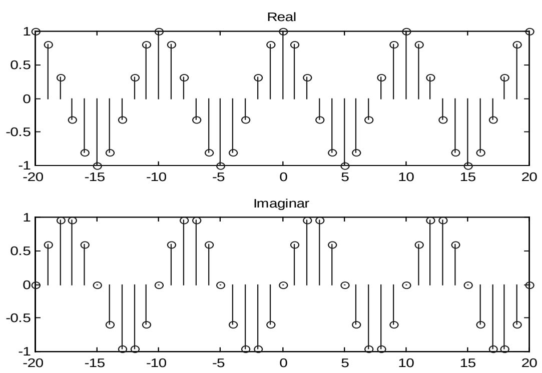
- ***Semnale complexe***

Să se definească secvenţele complexe:

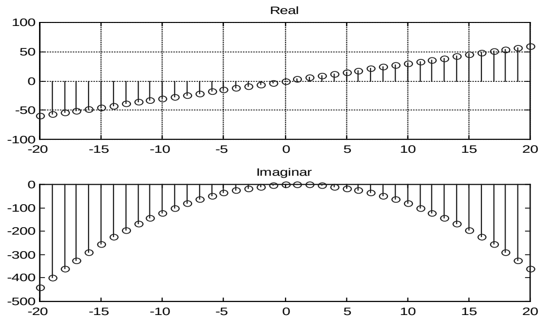
1. $x_1[n]=e^{jn\frac{\pi}{5}}$
2. $x_2[n]=3n-j(n-1)^2$ pentru $-20\le n\le 20$.

Să se reprezinte grafic părţile reale si imaginare ale acestor secvenţe.

```
n=-20:20;
x1=exp(j*n*pi/5);
subplot(2,1,1),stem(n,real(x1)),title('Real')
subplot(2,1,2),stem(n,imag(x1)),title('Imaginar')
```



```
x2=3*n-j*(n-1).^2;
subplot(2,1,1),stem(n,real(x2)),title('Real'),grid
subplot(2,1,2),stem(n,imag(x2)),title('Imaginar')
```



Explicaţi rezultatul obţinut în urma comenzilor:

```
plot(x1)
plot(x2)
```

// vezi secţiunea 1.2.12 din capitolul 1.

2.1.2. Convoluţia liniară a semnalelor în timp discret

Produsul de convoluţie liniară a două secvenţe numerice $x_1[n]$ şi $x_2[n]$ este definit astfel:

$$x[n] = x_1[n] * x_2[n] = \sum_{k=-\infty}^{\infty} x_1[k] \cdot x_2[n-k] \qquad (2.6)$$

unde secvenţele $x_1[n]$ şi $x_2[n]$ s-au presupus a avea lungime infinită.

În MATLAB secvenţele numerice sunt privite ca nişte vectori şi ţinând cont de faptul că ele sunt finite (suma nu va mai fi de la $-\infty$ la $+\infty$) şi că indicele primului element dintr-un vector nu poate fi zero (se efectuează o indexare cu 1) atunci definiţia de mai sus devine:

$$x[n+1] = \sum_{k=0}^{N-1} x_1[k+1] \cdot x_2[n-k] \qquad (2.7)$$

unde N este maximul dintre lungimile celor două secvenţe.

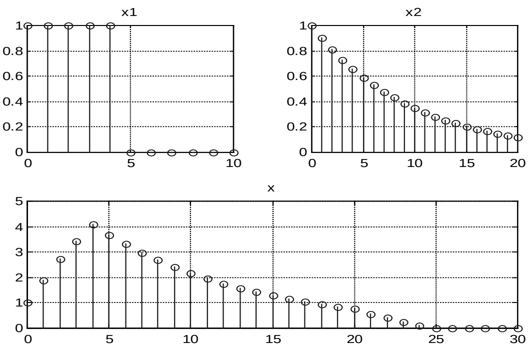
***Sintaxa*:**

`conv(x1,x2)`

- returnează ca rezultat un vector de *lungime egală cu lungimea vectorului* `x1` *plus lungimea vectorului* `x2` *minus* 1, ce reprezintă produsul de convoluţie liniară al celor două secvenţe definite prin vectorii `x1` şi `x2`.

***Exemplu*:**
Să se calculeze şi să se reprezinte grafic produsul de convoluție liniară a secvențelor:
1. $x_1[n] = u[n] - u[n-5]$ pentru $0 \le n \le 10$
2. $x_2[n] = (0{,}9)^n$ pentru $0 \le n \le 20$

```
x1=[ones(1,5),zeros(1,6)];
n=0:20;
x2=0.9.^n;
x=conv(x1,x2);
subplot(2,2,1),stem(0:10,x1),title('x1'),grid
subplot(2,2,2),stem(n,x2),title('x2'),grid
subplot(2,1,2),stem(0:length(x)-1,x),title('x'),grid
```



Explicați modul în care a fost divizată fereastra grafică prin utilizarea comenzii `subplot`.
Verificați relația dintre dimensiunile vectorilor `x1`, `x2` şi `x` folosind funcția `length`.

E5. ***Exerciții*:**
Să se calculeze şi să se reprezinte grafic (ca în exemplul precedent) produsul de convoluție liniară a următoarelor perechi de secvențe:

1. $x_1[n] = \delta[n]$ $\quad x_2[n] = (0{,}9)^n$ $\quad$ pentru $0 \le n \le 20$
2. $x_1[n] = \delta[n-5]$ $\quad x_2[n] = (0{,}9)^n$ $\quad$ pentru $0 \le n \le 20$
3. $x_1[n] = u[n]$ $\quad x_2[n] = |10 - n|$ $\quad$ pentru $0 \le n \le 20$
4. $x_1[n] = u[n-5]$ $\quad x_2[n] = |10 - n|$ $\quad$ pentru $0 \le n \le 20$
5. $x_1[n] = u[n-5] - u[n-15]$ $\quad x_2[n] = |10 - n|$ $\quad$ pentru $0 \le n \le 20$
6. $x_1[n] = \sin(0{,}11 \cdot n \cdot \pi)$ $\quad x_2[n] = (-1)^n$ $\quad$ pentru $0 \le n \le 20$
7. $x_1[n] = \cos(0{,}11 \cdot n \cdot \pi)$ $\quad x_2[n] = (-1)^n$ $\quad$ pentru $0 \le n \le 20$

Explicați rezultatele obținute.

2.1.3. Transformata Fourier discretă

Transformata Fourier în timp discret (DTFT – *Discrete Time Fourier Transform*) a unei secvenţe $x[n]$ este dată de relaţia:

$$X(e^{j\omega}) = \sum_{n} x[n] e^{-j\omega n} \tag{2.8}$$

unde ω este pulsaţia normată ($\omega = 2\pi f / F_{esantionare}$). Funcţia $X(e^{j\omega})$ este periodică de perioadă 2π, deci este suficient să cunoaştem comportarea sa în intervalul $[-\pi, \pi)$ (*interval de bază*). Datorită faptului că această funcţie este continuă, variabila ω putând lua o infinitate de valori, nu este posibilă o implementare pe o maşină de calcul. Pentru a realiza totuşi o analiză în frecvenţă se utilizează *transformata Fourier discretă* (DFT – *Discret Fourier Transform*), obţinută prin discretizarea variabilei ω pe intervalul $[0, 2\pi)$ în N puncte ($\omega \to 2k\pi / N$, cu $k = 0,1,\ldots,N-1$). Astfel, *transformata Fourier discretă* a unei secvenţe $x[n]$ este dată de relaţia:

$$X[k] = \sum_{n} x[n] e^{-j\frac{2k\pi}{N}n} \qquad \text{cu} \qquad k = 0,1,\ldots,N-1 \tag{2.9}$$

În MATLAB, pentru calculul transformatei Fourier discrete se foloseşte funcţia `fft`. Denumirea sa reprezintă prescurtarea de la *Fast Fourier Transform* (transformata Fourier rapidă) şi indică faptul că este folosit pentru calcul un algoritm rapid. Pentru obţinerea unor rezultate optime în cazul utilizării acestei funcţii se recomandă ca lungimea transformatei *N* să fie aleasă ca putere a lui 2 (exemplu: 256, 512, 1024 etc.).

***Sintaxe*:**

`y = fft(x)`

- <u>dacă x este un vector</u> se returnează un vector y de aceeaşi dimensiune cu vectorul x ce conţine valorile transformatei Fourier discrete aplicată elementelor vectorului x; lungimea transformatei Fourier (numărul de puncte *N* în care se calculează transformata) este egală în acest caz cu lungimea vectorului x.
- <u>dacă x este o matrice</u> se va returna matricea y de aceeaşi dimensiune cu matricea x; coloana *i* din matricea y va conţine valorile transformatei Fourier discrete aplicată elementelor coloanei *i* din matricea x.

`y = fft(x,N)`

- aceleaşi considerente ca în sintaxa precedentă cu deosebirea că în acest caz se specifică şi numărul de puncte *N* în care se calculează transformata.

***Exemple*:**

Să se calculeze transformata Fourier discretă a secvenţelor:

1. $x_1[n] = u[n] - u[n-10]$
2. $x_2[n] = \sin\left(\frac{n\pi}{5}\right)$
3. $x_3[n] = \cos\left(\frac{n\pi}{5}\right)$, pentru $0 \le n \le 20$.

Să se reprezinte grafic partea reală, partea imaginară, modulul şi faza transformatelor Fourier discrete calculate.

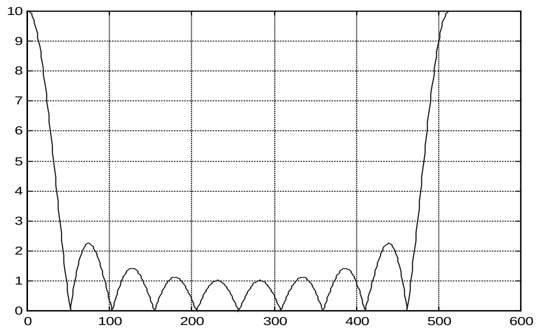
```
n=0:20;
x1=[ones(1,10),zeros(1,11)];
X=fft(x1);
plot(X)
```

// *explicaţi* rezultatul graficului obţinut (vezi secţiunea 1.2.12 din capitolul 1).

```
plot(abs(X))
```

// calculul transformatei Fourier discrete s-a efectuat într-un număr de puncte egal cu lungimea vectorului x1 (vezi sintaxa de la `fft`); pentru o mai bună reprezentare vom efectua calculul într-un număr mai mare de puncte (512):

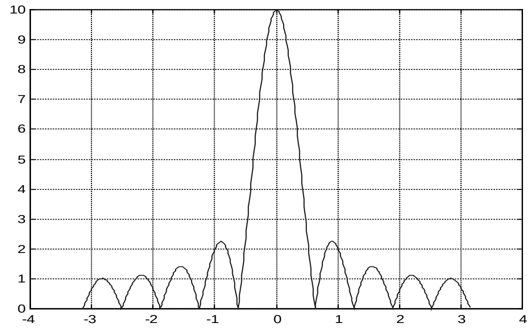
```
X1=fft(x1,512);
plot(abs(X1)),grid
```



// această reprezentare corespunde însă intervalului de frecvenţă $[0,2\pi)$ (pe abscisă avem numărul de puncte al vectorului X1 pentru că în sintaxa funcţiei `plot` nu s-a specifcat nimic altceva);

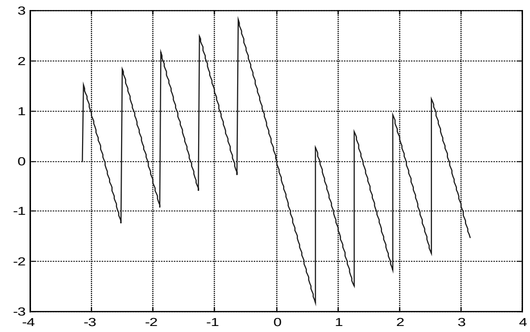
De obicei se doreşte însă reprezentarea în intervalul de bază $[-\pi,\pi)$. Având în vedere faptul că funcţia este periodică de perioadă 2π atunci reprezentarea din intervalul $[-\pi,0)$ corespunde cu reprezentarea din intervalul $[\pi,2\pi)$. Prin urmare trebuie realizată o inversare a celor două jumătăţi ale vectorului X1. Acest lucru se realizează în MATLAB prin utilizarea comenzii `fftshift` (schimbă între ele cele două jumătăţi ale unui vector). În plus, pentru a avea pe abscisă reprezentarea în intervalul $[-\pi,\pi)$ trebuie generat un vector cu pas liniar care să conţină în intervalul respectiv un număr de elemente egal cu lungimea transformatei Fourier discrete calculate (512).

```
w=-pi:2*pi/512:pi-2*pi/512;
plot(w,fftshift(abs(X1))),grid
```



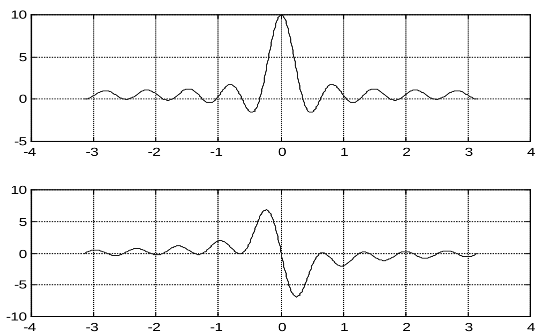
// s-a reprezentat astfel modulul transformatei Fourier discrete calculate, în intervalul de bază $[-\pi,\pi)$.

```
plot(w,fftshift(angle(X1))),grid
```



// s-a reprezentat faza transformatei Fourier discrete calculate.

```
subplot(2,1,1),plot(w,fftshift(real(X1))),grid
subplot(2,1,2),plot(w,fftshift(imag(X1))),grid
```

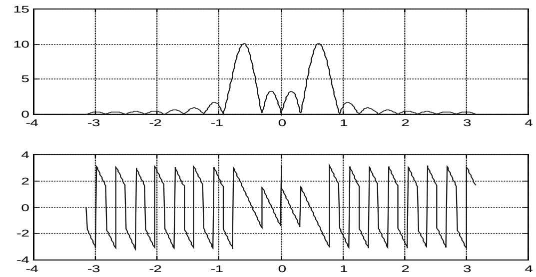


// s-au reprezentat partea reală şi partea imaginară a transformatei Fourier discrete calculate.

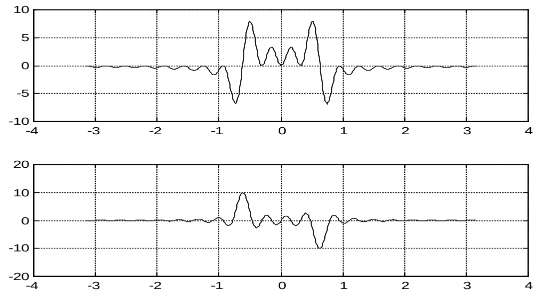
Explicaţi aliura caracteristicilor obţinute. *Observaţi* simetria pară sau impară a reprezentărilor, trecerile prin zero şi "salturile" din caracteristica de fază.

În mod asemănător se procedează şi pentru celelalte două secvenţe:

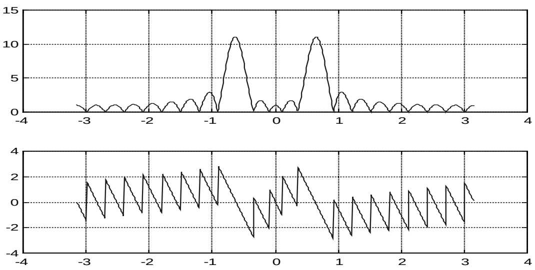
```
x2=sin(n*pi/5);
X2=fft(x2,512);
subplot(2,1,1),plot(w,fftshift(abs(X2))),grid
subplot(2,1,2),plot(w,fftshift(angle(X2))),grid
```



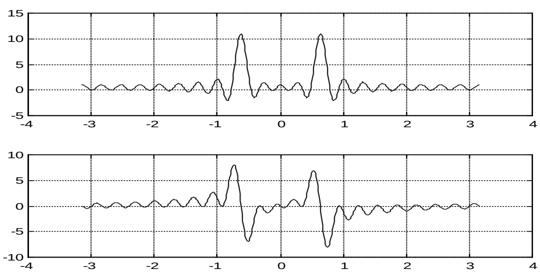
```
subplot(2,1,1),plot(w,fftshift(real(X2))),grid
subplot(2,1,2),plot(w,fftshift(imag(X2))),grid
```



```
x3=cos(n*pi/5);
X3=fft(x3,512);
subplot(2,1,1),plot(w,fftshift(abs(X3))),grid
subplot(2,1,2),plot(w,fftshift(angle(X3))),grid
```



```
subplot(2,1,1),plot(w,fftshift(real(X3))),grid
subplot(2,1,2),plot(w,fftshift(imag(X3))),grid
```



Explicaţi aliura caracteristicilor obţinute.

E6. ***Exerciții*:**

Să se calculeze transformata Fourier discretă a secvențelor:

1. $x_1[n] = \delta[n]$ pentru $0 \le n \le 10$.

2. $x_2[n] = \delta[n-1] - \delta[n-4] + \delta[n-8]$ pentru $0 \le n \le 10$.

3. $x_3[n] = n$ pentru $0 \le n \le 5$.

4. $x_4[n] = e^{-j\frac{n\pi}{3}} u[n-2]$ pentru $0 \le n \le 15$.

5. $x_5[n] = \begin{cases} n & ,0 \le n \le 5 \\ 10-n & ,6 \le n \le 10 \end{cases}$

6. $x_6[n] = \dfrac{2}{n^2+1} e^{-j\frac{n\pi}{5}}$ pentru $0 \le n \le 20$.

7. $x_7[n] = e^{-j\frac{n\pi}{5}} + e^{-j\frac{n\pi}{3}}$ pentru $0 \le n \le 20$.

8. $x_8[n] = \dfrac{1}{n+1} \ln(3n+5)$ pentru $0 \le n \le 10$.

9. $x_9[n] = \dfrac{3}{5} \cdot \dfrac{\sin(0.6 * pi * n - 20)}{0.6 * pi * n - 10}$ pentru $0 \le n \le 20$.

Să se reprezinte grafic partea reală, partea imaginară, modulul şi faza transformatelor Fourier discrete calculate.
Pentru toate secvențele verificați relația lui Parseval:

$$\sum_n |x[n]|^2 = \frac{1}{N} \sum_k |X[k]|^2 \tag{2.10}$$

Alegeți oricare două secvențe şi verificați proprietatea de liniaritate a transformatei Fourier discrete:

$$DFT\{ax[n] + by[n]\} = a \cdot DFT\{x[n]\} + b \cdot DFT\{y[n]\} \tag{2.11}$$

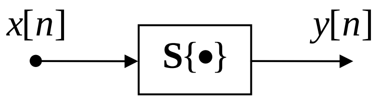
alegând două valori oarecare pentru constantele *a* şi *b*.

2.2. Sisteme în timp discret

Un sistem în timp discret transformă secvenţa de intrare $x[n]$ într-o altă secvenţă de ieşire $y[n]$. Notând $\mathbf{S}\{\bullet\}$, operatorul sistemului, acesta este descris matematic prin relaţia:

$$y[n] = \mathbf{S}\{x[n]\} \tag{2.12}$$

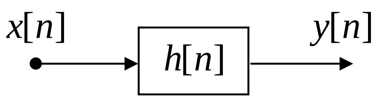
Reprezentarea simbolică a unui sistem în timp discret este următoarea:



Pentru un sistem în timp discret, liniar şi invariant în timp (*SDLIT*) se numeşte *funcţie pondere* sau *răspuns la impulsul unitate*, secvenţa care se obţine la ieşirea sistemului dacă la intrare s-a aplicat impulsul unitate $\delta[n]$:

$$h[n] = \mathbf{S}\{\delta[n]\} \tag{2.13}$$

Astfel, un alt mod de a reprezenta simbolic un sistem în timp discret este:



Răspunsul $y[n]$ al unui *SDLIT* la orice secvenţă $x[n]$, poate fi determinat prin convoluţie dacă se cunoaşte răspunsul la impulsul unitate $h[n]$:

$$y[n] = x[n] * h[n] = \sum_k x[k]h[n-k] \tag{2.14}$$

SDLIT pot fi reprezentate prin *ecuaţii cu diferenţe finite* cu coeficienţi constanţi, care dau legătura între secvenţa de intare şi cea de ieşire:

$$\sum_{k=0}^{N} a_k y[n-k] = \sum_{k=0}^{M} b_k x[n-k] \tag{2.15}$$

Pentru analiza în frecvenţă a *SDLIT* reamintim formula pentru *transformata **Z*** a unei semnal în timp discret $x[n]$:

$$X(z) = \sum_n x[n] z^{-n} \tag{2.16}$$

Pentru un *SDLIT* se numeşte *funcţie de sistem* sau *funcţie de transfer* $H(z)$, raportul între transformatele **Z** ale secvenţei de ieşire (răspunsul $y[n]$) şi secvenţei de intrare (excitaţia $x[n]$), reprezentând de fapt transformata **Z** a funcţiei pondere $h[n]$:

$$H(z) = \frac{Z\{y[n]\}}{Z\{x[n]\}} = \frac{Y(z)}{X(z)} = Z\{h[n]\} \tag{2.17}$$

Ţinând cont de ecuaţia cu diferenţe finite şi de proprietatea de întârziere a transformatei **Z** ($\mathbf{Z}\{x[n-k]\} = z^{-k}X(z)$), funcţia de sistem $H(z)$ se mai poate scrie astfel:

$$H(z) = \frac{\sum_{k=0}^{M} b_k z^{-k}}{1 + \sum_{k=1}^{N} a_k z^{-k}} \tag{2.18}$$

în care s-a presupus că $a_0 = 1$. Rădăcinile polinomului de la numărător se numesc *zerourile* funcţiei de transfer iar rădăcinile polinomului de la numitor se numesc *polii* funcţiei de transfer. Pentru ca sistemul să fie *stabil* polii trebuie să fie situaţi în interiorul cercului de rază unitate (modul lor să fie subunitar).

2.2.1. Răspunsul la impuls al unui *SDLIT*

Funcţia MATLAB `impz` permite determinarea şi afişarea răspunsului la impuls (funcţia pondere) $h[n]$ a unui *SDLIT* dacă se cunosc coeficienţii a_k şi b_k din ecuaţia cu diferenţe finite sau din expresia funcţiei de transfer $H(z)$.

***Sintaxe*:**

`[h,t] = impz(b,a)`

- vectorul `b` conţine coeficienţii b_k (`b` = [$b_0, b_1, ..., b_M$]) iar vectorul `a` conţine coeficienţii a_k (`a` = [1, $a_1, a_2, ..., a_N$]); se vor returna un *vector coloană* `h` care va conţine valorile eşantioanelor răspunsului la impuls al sistemului, $h[n]$, şi un *vector coloană* `t` ce va conţine momentele de pe axa timp (abscisa), alese în mod implicit, în care au fost calculate valorile eşantioanelor; *parametrul de ieşire* `t` *poate să lipsească din sintaxă în cazul în care ne interesează doar răspunsul la impuls* `h`.

`[h,t] = impz(b,a,n)`

- se precizează numărul de puncte `n` în care se doreşte a fi calculat răspunsul la impuls `h`; vectorul coloană `t` va conține valorile acestor puncte (acestea vor fi 0, 1, 2, ..., n-1); *parametrul de ieşire* `t` *poate să lipsească din sintaxă în cazul în care ne interesează doar răspunsul la impuls* `h`.

`[h,t] = impz(b,a,n,Fs)`

- aceleaşi considerente ca în sintaxa precedentă cu deosebirea că cele `n` valori din vectorul `t` vor fi distanțate cu pasul 1/Fs (0, 1/Fs, 2/Fs, ..., (n-1)/Fs).

`[h,t] = impz(b,a,[],Fs)`

- aceleaşi considerente ca în sintaxa precedentă cu deosebirea că se alege în mod implicit numărul de puncte în care se calculează răspunsul la impuls `h`.

`impz(b,a,...)`

- reprezintă grafic răspunsul la impuls calculat; punctele de suspensie au fost introduse pentru a sugera faptul că se poate folosi oricare combinație a parametrilor de intrare din sintaxele precedente.

***Exemple*:**
Să se determine şi să se reprezinte grafic funcția pondere a unui *SDLIT* definit prin:

1. $y[n]-0{,}9y[n-1]=0{,}3x[n]+0{,}6x[n-1]+0{,}6x[n-2]$
2. $H(z)=\dfrac{1+0{,}5z^{-1}}{1-1{,}8\cos(\pi/16)z^{-1}+0{,}81z^{-2}}$

```
b=[0.3,0.6,0.6];
a=[1,-0.9];
```

// s-au definit vectorii `b` şi `a` ce conțin valorile coeficienților b_k şi a_k din ecuația cu diferențe finite.

```
[h,t]=impz(b,a);
```

// s-a calculat răspunsul la impuls al sistemului.

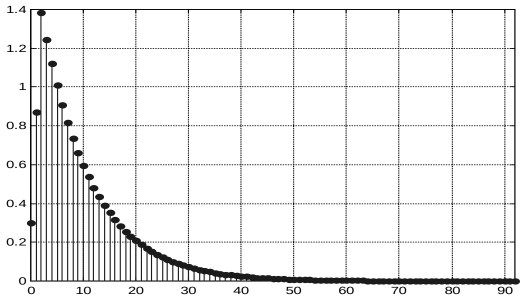
```
size(h)       →         ans =
                                  93      1
size(t)       →         ans =
                                  93      1
```

// se verifică faptul că `h` şi `t` sunt vectori coloană (vezi sintaxa) cu 93 de elemente fiecare.

Se poate verifica, tastând în fereastra de comenzi `t` urmat de *enter*, că valorile vectorului t sunt 0, 1, …, 92.

Pentru reprezentarea grafică se poate proceda în două moduri:

```
impz(b,a),grid
```

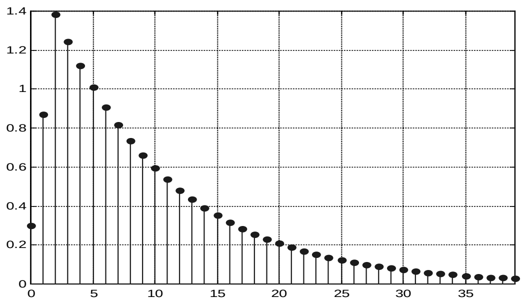



```
stem(t,h),grid
```

// se va obţine acelaşi rezultat grafic.

Dacă dorim afişarea numai a primelor 40 de eşantioane ale răspunsului la impuls procedăm în felul următor:

```
impz(b,a,40),grid
```



Verificaţi şi *explicaţi* efectul comenzii:

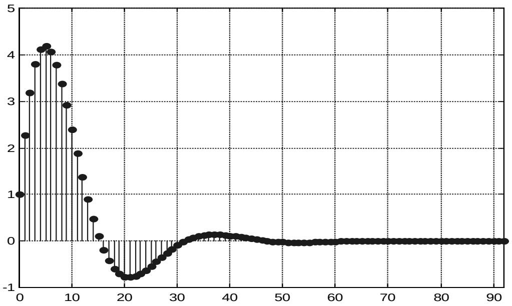
```
impz(b,a,40,3),grid
```

În mod asemănător procedăm şi pentru cel de al doilea exemplu:

```
b=[1,0.5];
a=[1,-1.8*cos(pi/16),0.81];
h=impz(b,a);
```

// vectorul coloană `h` va contine valorile eşantioanelor răspunsului la impuls al sistemului definit prin funcţia de transfer $H(z)$ din exemplu.

```
impz(b,a),grid
```



Verificaţi şi *explicaţi* efectul comenzii:

```
stem(0:length(h)-1,h),grid
```

E7. ***Exerciţii*:**
Să se determine şi să se reprezinte grafic funcţia pondere a unui sistem definit prin:
1. $y[n] = x[n] - 1{,}27x[n-1] + 0{,}81x[n-2] - 0{,}5x[n-3] + 0{,}125x[n-4]$
2. $y[n] + 0{,}9y[n-1] = x[n]$
3. $y[n] + 0{,}13y[n-1] + 0{,}52y[n-2] + 0{,}3y[n-3] =$
   $= 0{,}16x[n] - 0{,}48x[n-1] + 0{,}48x[n-2] - 0{,}16x[n-3]$
4. $y[n] + 0{,}9y[n-2] = 0{,}3x[n] + 0{,}6x[n-1] + 0{,}3x[n-2]$
5. $H(z) = \dfrac{1 - 0{,}5z^{-1} + 0{,}125z^{-2} - 0{,}075z^{-3}}{1 - 0{,}8z^{-1} + 0{,}64z^{-2} - 0{,}4z^{-3} + 0{,}024z^{-4}}$
6. $H(z) = \dfrac{1}{1 - 0{,}77z^{-1} + 0{,}44z^{-3}}$
7. $H(z) = 1 - 1{,}27z^{-1} + 0{,}81z^{-2} - 0{,}5z^{-3} + 0{,}125z^{-4} - 0{,}3z^{-7}$
8. $H(z) = \dfrac{1}{1 - z^{-5}}$

2.2.2. Răspunsul unui *SDLIT* la un semnal de intrare

Funcţia MATLAB `filter` permite determinarea răspunsului $y[n]$ al unui *SDLIT* dacă se cunosc coeficienţii a_k şi b_k din ecuaţia cu diferenţe finite sau din expresia funcţiei de transfer $H(z)$ şi semnalul de intrare în sistem $x[n]$.

***Sintaxe*:**
`y = filter(b,a,x)`
- dacă `x` este un vector atunci se returnează un vector `y` de aceeaşi dimensiune cu vectorul `x`; vectorul `x` conţine valorile semnalului de intrare în filtru (excitaţia); vectorul `b` conţine coeficienţii b_k (`b` = [b_0, b_1, ..., b_M]) iar vectorul `a` conţine coeficienţii a_k (`a` = [1, a_1, a_2, ..., a_N]); *dacă primul element din vectorul* `a` *este diferit de* 1 *atunci funcţia* `filter` *normează coeficienţii* a_k *ai sistemului la valoarea primului element din vectorul* `a` (*astfel primul element din vectorul* `a` *devine* 1); vectorul obţinut `y` reprezintă răspunsul sistemului definit de coeficienţii din vectorii `b` şi `a`, dacă la intrare este aplicată secvenţa definită de vectorul `x`.
- dacă `x` este o matrice atunci se returnează o matrice `y` de aceeaşi dimensiune cu matricea `x`; funcţia `filter` va opera în acest caz pe coloane: coloana *k* din matricea `y` reprezintă răspunsul sistemului definit de coeficienţii din vectorii `b` şi `a`, dacă la intrare este aplicată coloana *k* din matricea `x`.

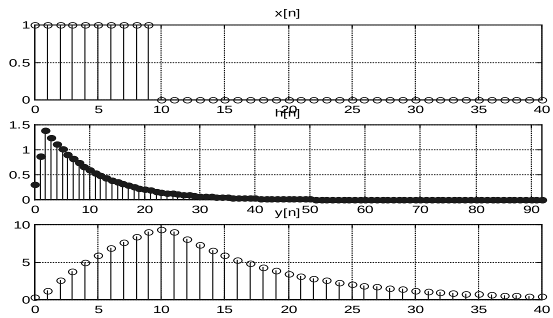
***Exemple*:**
Să se determine şi să se reprezinte grafic răspunsul unui sistem definit prin:

1. $y[n]-0{,}9y[n-1]=0{,}3x[n]+0{,}6x[n-1]+0{,}6x[n-2]$

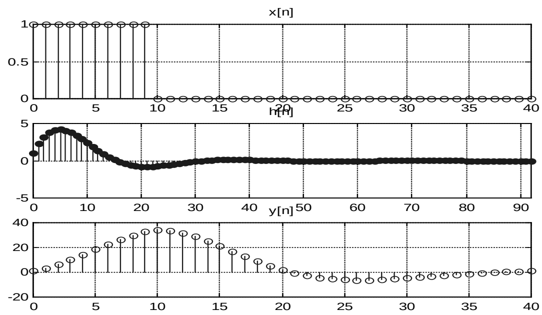
2. $H(z)=\dfrac{1+0{,}5z^{-1}}{1-1{,}8\cos(\pi/16)z^{-1}+0{,}81z^{-2}}$

la semnalul de intrare:
$x_1[n]=u[n]-u[n-10]$, pentru $0\le n\le 40$.

```
b=[0.3,0.6,0.6];
a=[1,-0.9];
```
// vectorii `b` şi `a` conţin valorile coeficienţilor b_k şi a_k.
```
x=[ones(1,10),zeros(1,31)];
```
// s-a definit vectorul `x` corespunzător secvenţei de intrare.
```
y=filter(b,a,x);
```
// s-a calculat răspunsul sistemului la secvenţa de intrare definită prin vectorul `x`.
Verificaţi faptul că vectorii `y` şi `x` au aceeaşi dimensiune (folosind `size`).
```
n=0:40;
subplot(3,1,1),stem(n,x),grid,title('x[n]')
subplot(3,1,2),impz(b,a),grid,title('h[n]')
subplot(3,1,3),stem(n,y),grid,title('y[n]')
```



În mod asemănător procedăm şi pentru cel de al doilea exemplu:
```
b=[1,0.5];
a=[1,-1.8*cos(pi/16),0.81];
y=filter(b,a,x);
subplot(3,1,1),stem(n,x),grid,title('x[n]')
subplot(3,1,2),impz(b,a),grid,title('h[n]')
subplot(3,1,3),stem(n,y),grid,title('y[n]')
```



E8. ***Exerciţii*:**
Să se determine răspunsul sistemelor definite prin relaţiile de la punctele 1–8 din cadrul exerciţiilor de la funcţia `impz` (vezi secţiunea 2.2.1., pagina 42) la următoarele semnale de intrare:

1. $x_1[n] = \delta[n]$ pentru $0 \le n \le 40$
2. $x_2[n] = u[n]$ pentru $0 \le n \le 40$
3. $x_3[n] = n$ pentru $0 \le n \le 5$ (6 perioade din $x_3[n]$)
4. $x_4[n] = \begin{cases} n & ,0 \le n \le 10 \\ 20 - n & ,11 \le n \le 20 \end{cases}$
5. $x_5[n] = \sin\left(\dfrac{n\pi}{5}\right)$ pentru $0 \le n \le 20$
6. $x_6[n] = \cos\left(\dfrac{n\pi}{5}\right)$ pentru $0 \le n \le 20$

Să se reprezinte grafic (folosind `subplot` ca în exemplele precedente) semnalul de intrare, funcţia pondere a sistemului şi semnalul de ieşire.

2.2.3. Răspunsul în frecvenţă al *SDLIT*

Funcţia MATLAB `freqz` permite determinarea răspunsului în frecvenţă al unui *SDLIT* dacă se cunosc coeficienţii a_k şi b_k din ecuaţia cu diferenţe finite sau din expresia funcţiei de transfer $H(z)$. Dacă este apelată fără parametrii de ieşire, cum se va vedea în sintaxă, această funcţie reprezintă grafic caracteristicile de amplitudine-frecvenţă şi fază-frecvenţă ale *SDLIT* respectiv.

***Sintaxe*:**

`[H,W] = freqz(b,a,n)`

- vectorul `b` conţine coeficienţii b_k (`b` = [b_0, b_1, ..., b_M]), vectorul `a` conţine coeficienţii a_k (`a` = [1, a_1, a_2, ..., a_N]), iar `n` reprezintă numărul de puncte în care se calculează răspunsul în frecvenţă `H`; vectorul `W` va conţine valorile acestor `n` puncte (valorile vor fi cuprinse între 0 şi π); este recomandat să se aleagă `n` putere a lui 2 (pentru a permite un calcul eficient folosind un algoritm FFT rapid); *dacă* `n` *nu se specifică se alege în mod implicit* 512.

`[H,F] = freqz(b,a,n,Fs)`

- această sintaxă permite specificarea unei valori pentru frecvenţa de eşantionare `Fs` (în Hz); vectorul `F` va conţine valorile celor `n` puncte în care se calculează răspunsul în frecvenţă `H` (în acest caz valorile acestor puncte vor fi cuprinse între 0 şi `Fs`/2);

`[H,W] = freqz(b,a,n,'whole')`

- aceleaşi considerente ca în cazul primei sintaxe cu deosebirea că valorile celor `n` puncte de calcul, conținute în vectorul `W`, vor fi cuprinse între 0 şi 2π; *dacă* `n` *nu se specifică se alege în mod implicit* 512.

`[H,F] = freqz(b,a,n,'whole',Fs)`

- aceleaşi considerente ca în cazul celei de a doua sintaxe cu deosebirea că valorile celor `n` puncte de calcul, conținute în vectorul `F`, vor fi cuprinse între 0 şi `Fs`; *dacă* `n` *nu se specifică se alege în mod implicit* 512.

`H = freqz(b,a,W)`

- răspunsul în frecvență `H` se calculează la frecvențele specificate în vectorul `W`; valorile acestor frecvențe trebuie să fie cuprinse între 0 şi 2π; *dacă* `W` *nu se specifică se aleg în mod implicit* 512 *valori de frecvență.*

`H = freqz(b,a,F,Fs)`

- răspunsul în frecvență `H` se calculează la frecvențele specificate în vectorul `F`; valorile acestor frecvențe trebuie să fie cuprinse între 0 şi `Fs` (frecvența de eşantionare în Hz).

`freqz(b,a,...)`

- reprezintă grafic caracteristicile amplitudine-frecvență şi fază-frecvență ale răspunsului în frecvență calculat; punctele de suspensie au fost introduse pentru a sugera faptul că se poate folosi oricare combinație a parametrilor de intrare din sintaxele precedente.

***Exemple*:**
Să se determine răspunsul în frecvență al *SDLIT* definite prin:

1. $y[n]+0{,}9y[n-1]=0{,}3x[n]+0{,}6x[n-1]+0{,}3x[n-2]$
2. $H(z)=\dfrac{0.634-0.634z^{-2}}{1-0.268z^{-2}}$

Să se reprezinte grafic caracteristicile amplitudine-frecvență şi fază-frecvență ale răspunsului în frecvență calculat.

```
b=[0.3,0.6,0.3];
a=[1,0.9];
[H,W]=freqz(b,a);
```

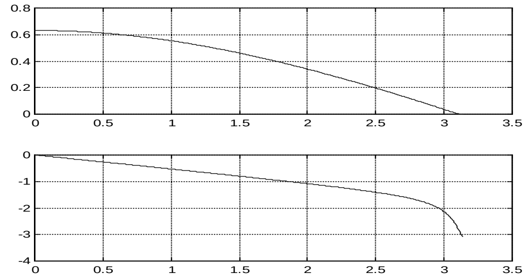
// s-a calculat răspunsul în frecvență `H` în 512 puncte de frecvență cuprinse în intervalul $[0,\pi]$.

Verificați dimensiunile vectorilor `H` şi `W`.

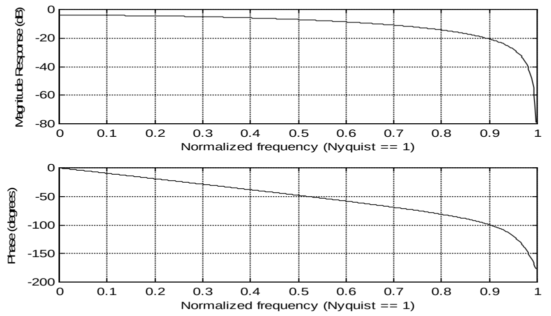
Verificați şi *explicați* rezultatul comenzii: `plot(H)`

Reprezentările grafice cerute se pot realiza în două moduri:

```
figure(1)
subplot(2,1,1),plot(W,abs(H)),grid
subplot(2,1,2),plot(W,angle(H)),grid
```



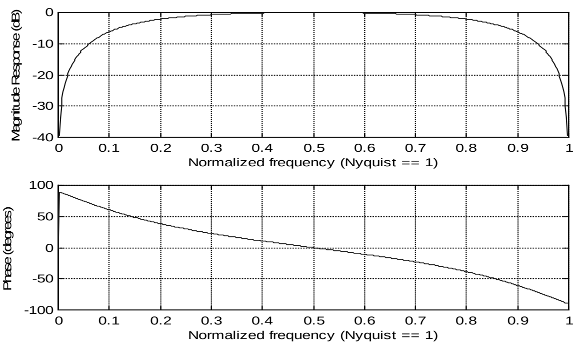
```
figure(2)
freqz(b,a)
```



Explicați diferența dintre cele două moduri de reprezentare.

În mod asemănător procedăm şi pentru cel de al doilea exemplu:

```
b=[0.634,0,-0.634];
a=[1,0,-0.268];
[H,W]=freqz(b,a);
freqz(b,a)
```



E9. ***Exerciții*:**

Să se determine răspunsul în frecvență al sistemelor definite prin relațiile de la punctele 1–8 din cadrul exercițiilor de la funcția `impz` (vezi secțiunea 2.2.1., pagina 42). Să se reprezinte grafic caracteristicile amplitudine-frecvență şi fază-frecvență pentru fiecare răspuns în frecvență calculat.

2.2.4. Diagrama poli-zerouri pentru funcția de sistem a unui *SDLIT*

Funcția MATLAB `zplane` permite afişarea diagramei poli-zerouri în cazul funcției de sistem a unui *SDLIT* dacă se cunosc valorile polilor şi zerourilor *sau* dacă se cunosc doar coeficienții a_k şi b_k din ecuația cu diferențe finite sau din expresia funcției de transfer $H(z)$.

***Sintaxe*:**

`zplane(z,p)`

- <u>dacă `z` şi `p` sunt doi vectori coloană</u> ce conțin valorile zerourilor şi respectiv polilor funcției de transfer $H(z)$ atunci se va afişa diagrama poli-zerouri, marcând zerourile cu semnul 'o' iar polii cu semnul 'x'; dacă există poli sau zerouri multiple, acestea vor avea înscris, lângă semnul respectiv şi ordinul de multiplicitate.
- <u>dacă z şi p sunt două matrice</u> afişarea diagramei poli-zerouri se va face pentru fiecare coloană în parte cu culori diferite.

`zplane(b,a)`

- <u>dacă `b` şi `a` sunt doi vectori linie</u> ce conțin valorile coeficienților b_k şi a_k atunci se va afişa diagrama poli-zerouri a funcției de transfer $H(z)$ (calculându-se rădăcinile polinoamelor de la numărătorul şi numitorul funcției de sistem).

Reprezentarea polilor şi zerourilor unei funcții de sistem (diagrama poli-zerouri) se face în *planul Z*, în raport cu cercul de rază unitate, având pe abscisă partea reală şi pe ordonată partea imaginară. Fiind privite deci ca numere complexe, valorile respective pot fi exprimate în *formă polară* sau *formă carteziană*.

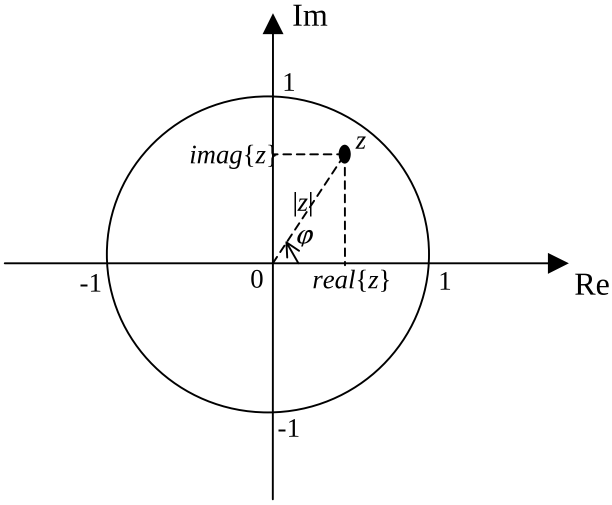
Orice număr complex z poate fi exprimat în formă polară astfel:

$$z = |z| \cdot e^{j\varphi} \tag{2.19}$$

în care: - $|z|$ = modulul numărului complex z ;
- φ = faza numărului complex z ;

În formă carteziană numărul complex z se exprimă sub forma:

$$z = real(z) + j \cdot imag(z) \tag{2.20}$$



Reprezentarea în planul Z a unui număr complez z

***Atenție*:**

În cazul în care dispunem de valorile coeficienților b_k şi a_k şi dorim să determinăm valorile polilor şi zerourilor funcției de sistem respective se poate utiliza funcția MATLAB `roots`, care calculează rădăcinile unui polinom dacă sunt precizați coeficienții acestuia.

Dacă se doreşte determinarea valorilor coeficienților b_k şi a_k, având valorile polilor şi zerourilor funcției de sistem, putem utiliza funcția MATLAB `poly`, care calculează coeficienții unui polinom dacă sunt precizate rădăcinile acestuia.

Verificați sintaxele celor două funcții folosind comanda `help`.

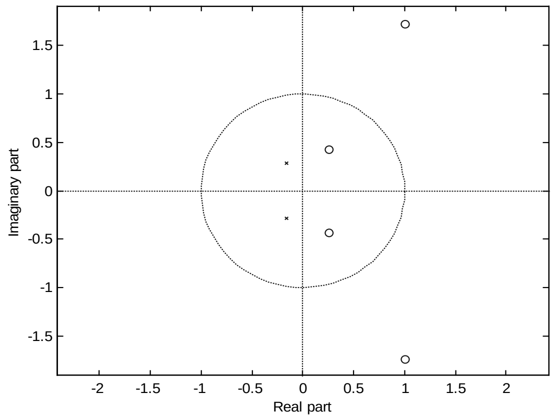
***Exemple*:**

1. Funcția de transfer a unui *SDLIT* are un zerou de valoare $r=\frac{1}{2}e^{j\frac{\pi}{3}}$. Ştiind că această funcție are zerouri şi în r^*, $\frac{1}{r}$, $\frac{1}{r^*}$ şi are doi poli în $q=\frac{1}{3}e^{j\frac{2\pi}{3}}$ şi q^* să se reprezinte diagrama poli-zerouri şi să se scrie forma nefactorizată a funcției de transfer (să se găsească valorile coeficienților b_k şi a_k).

```
r=1/2*exp(j*pi/3);
q=1/3*exp(j*2*pi/3);
z=[r;conj(r);1/r;1/conj(r)];
p=[q;conj(q)];
```

// s-au definit vectorii *coloană* `z` şi `p` (vezi sintaxa) ce conțin valorile zerourilor şi respectiv polilor funcției de transfer.

```
zplane(z,p)
```



// s-a reprezentat diagrama poli-zerouri.

```
b=poly(z)
     →   b =
            1.0000  -2.5000  5.2500  -2.5000  1.0000
a=poly(p)
     →   a =
            1.0000  0.3333  0.1111
```

// s-au calculat valorile coeficienţilor b_k şi a_k ai funcţiei de sistem. Forma nefactorizată a acestei funcţii va fi:

$$H(z) = \frac{1 - 2{,}5z^{-1} + 5{,}25z^{-2} - 2{,}5z^{-3} + z^{-4}}{1 - 0{,}3333z^{-1} + z^{-2}}$$

Verificaţi şi *explicaţi* rezultatul următoarelor comenzi:

```
roots(b)
roots(a)
zplane(b,a)
```

2. Să se reprezinte diagramele poli-zerouri pentru sistemele în timp discret definite prin:

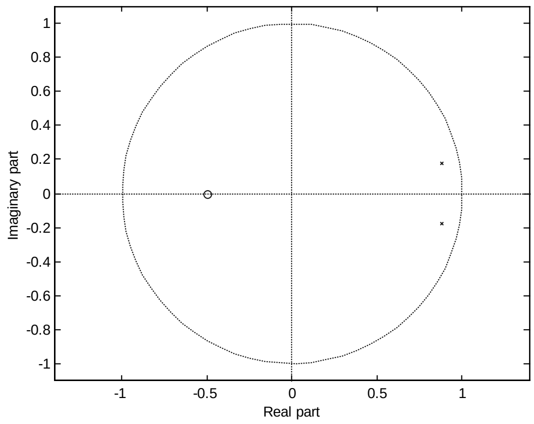
a. funcţia de transfer: $H(z) = \dfrac{1 + 0{,}5z^{-1}}{1 - 1{,}8\cos(\pi/16)z^{-1} + 0{,}81z^{-2}}$

b. funcţia de transfer: $H(z) = \dfrac{(1 + 0{,}5z^{-1})^2}{1 - 1{,}8\cos(\pi/16)z^{-1} + 0{,}81z^{-2}}$

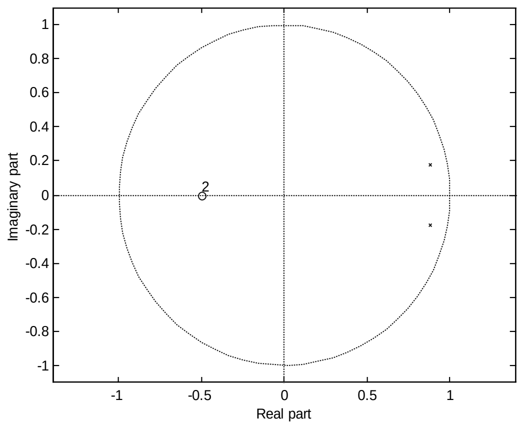
c. ecuaţia cu diferenţe finite:
$y[n] + 0.1y[n-1] + 0.1y[n-3] + 0.1y[n-4] = 0{,}3x[n] + 0{,}6x[n-1] + 0{,}6x[n-2]$

d. funcţia pondere: $h[n] = u[n]$, pentru $0 \le n \le 10$.

```
b=[1,0.5];
a=[1,-1.8*cos(pi/16),0.81];
zplane(b,a)
```

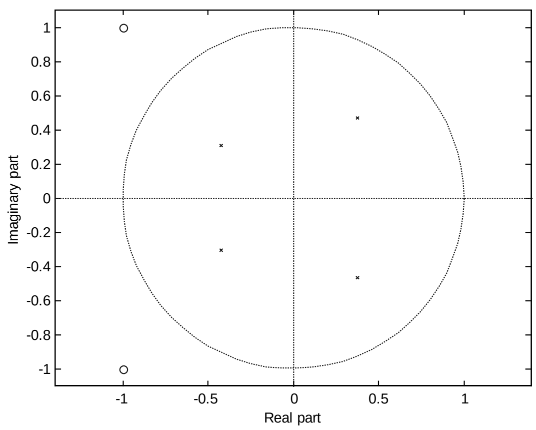



```
b=[1,1,0.25];
zplane(b,a)
```

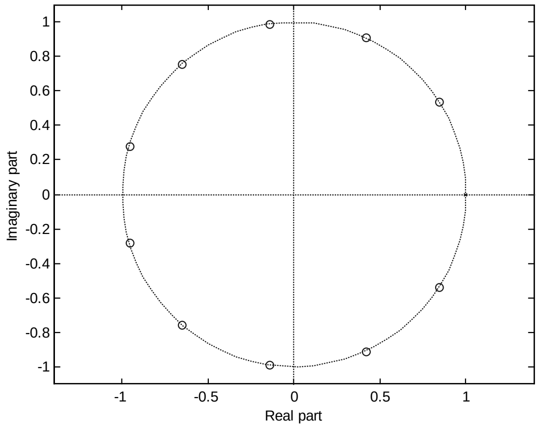


// se observă ordinul de multiplicitate ale zeroului.

```
b=[0.3,0.6,0.6];
a=[1,0.1,0,0.1,0.1];
zplane(b,a)
```



```
h=ones(1,11);
zplane(h,1)
```



Verificați şi *explicați* efectul comenzilor:

```
zplane(h)
zplane(h')
```

E10. ***Exerciții*****:**

1. Să se reprezinte diagramele poli-zerouri pentru sistemele definite prin relațiile de la punctele 1–8 din cadrul exercițiilor de la funcția `impz` (vezi secțiunea 2.2.1., pagina 42).
2. Se dă funcția de transfer a unui sistem în timp discret:

$$H(z)=\frac{(z-0{,}5e^{-j\pi/3})(z-2e^{-j\pi/3})(z-0{,}5e^{j\pi/3})(z-2e^{j\pi/3})}{(z-0{,}3e^{-j\pi/3})(z-0{,}3e^{j\pi/3})(z-0{,}7e^{-j\pi/5})(z-0{,}7e^{j\pi/5})}$$

   Să se reprezinte diagrama poli-zerouri şi să se scrie forma nefactorizată a funcției de transfer (să se găsească valorile coeficienților b_k şi a_k).