

Laborator 12

Funcții utilizate:

- **hist(x)** (sau **histogram**) - creează o histogramă cu bare, bazându-se pe elementele din vectorul dat ca parametru. Se împarte intervalul $[\min(x), \max(x)]$ în 10 părți egale. **hist** va afișa o serie de dreptunghiuri, iar înălțimea unui astfel de dreptunghi este defapt numărul de elemente din intervalul corespunzător (din cele 10 intervale identificate — — — > fiecărui dreptunghi îi corespunde un interval).

- **hist(x,nbins)** - prin variabila **nbins** specific în câte dreptunghiuri sau bin-uri se împart numerele din vectorul **x**, adică precizez numărul de subintervale.

- **hist(x,xbins)** - împarte elementele din **x** în subintervale în funcție de elementele din vectorul **xbins**. Dacă elementele din **xbins** sunt echidistante atunci **hist** folosește aceste valori ca centre pentru bin-uri, iar în caz contrar când nu sunt echidistante se consideră mijloacele subintervalelor date de 2 elemente consecutive din **xbins** ca vârfuri sau limite pentru bin-uri.

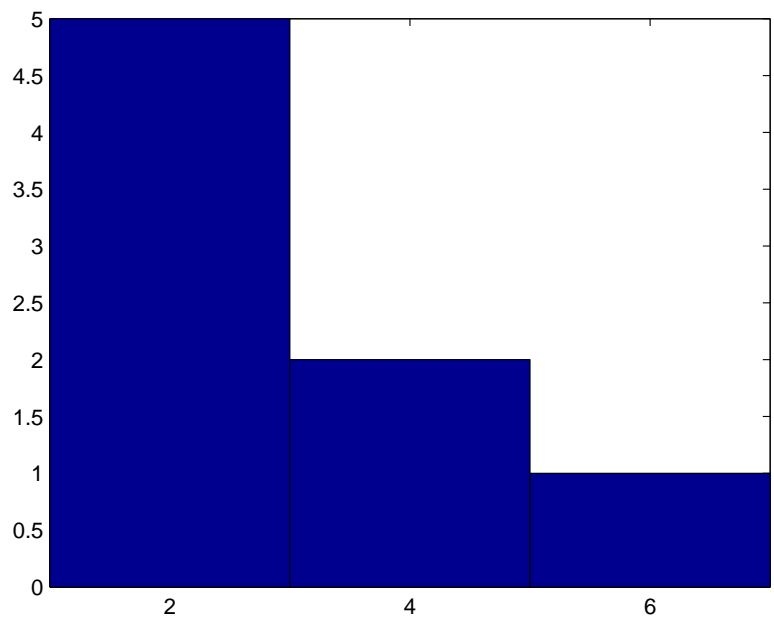
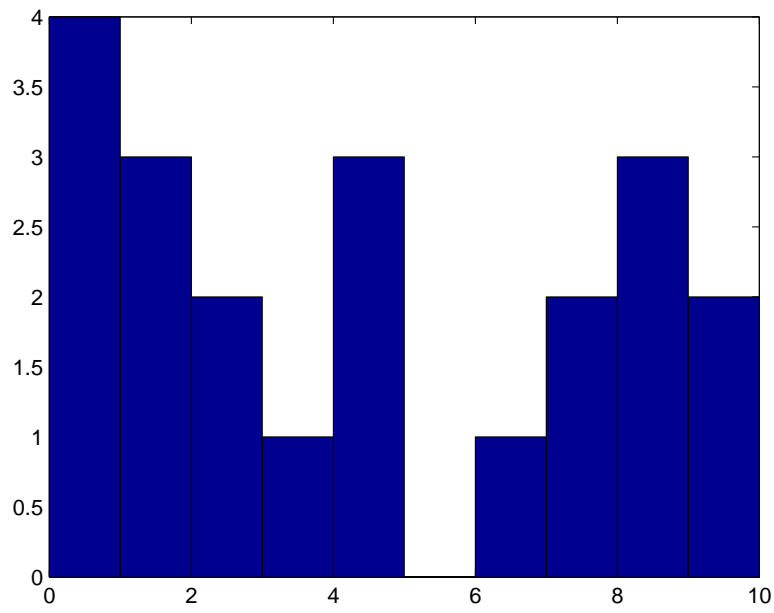
Exemple:

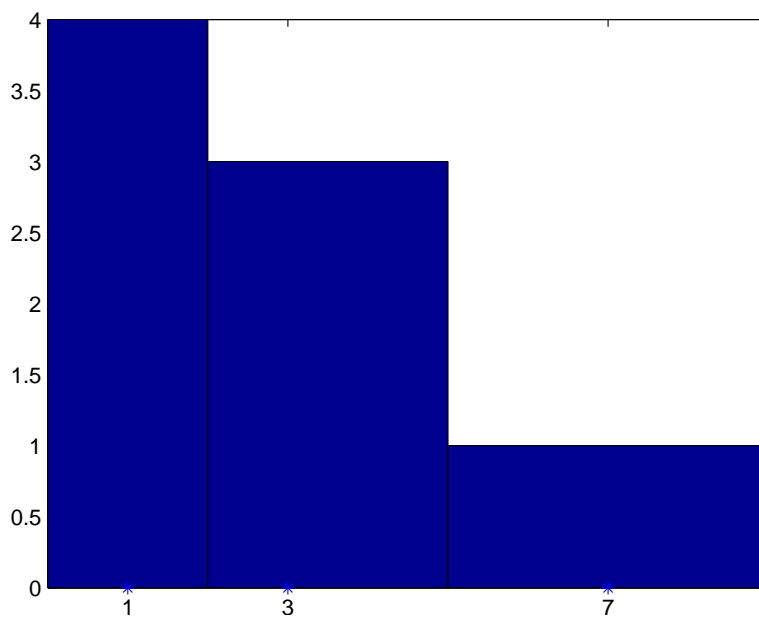
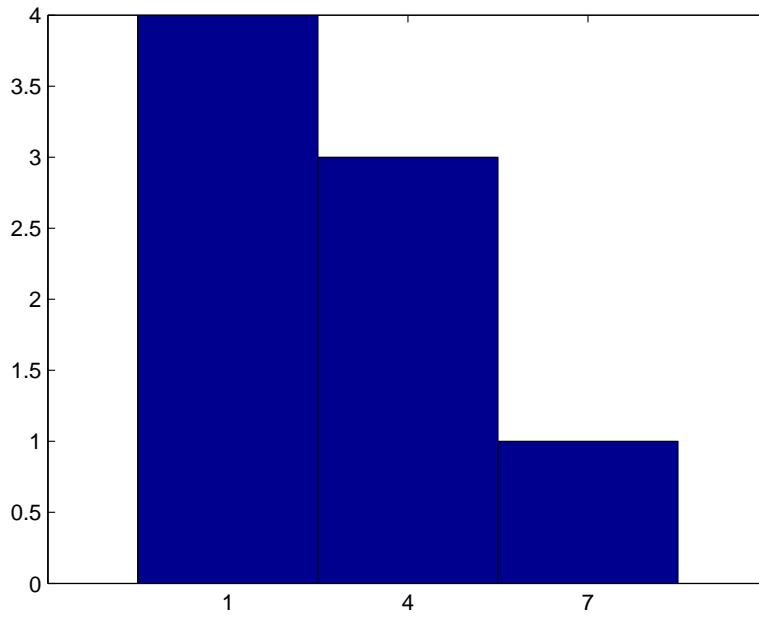
```
x = [0 2 9 2 5 8 7 3 1 9 4 3 5 8 10 0 1 2 9 5 10];  
figure(1)  
hist(x)
```

```
x1=[2 2 3 4 2 1 5 7];  
figure(2)  
hist(x1,3)
```

```
figure(3)  
hist(x1,[1 4 7])
```

```
figure(4)  
hist(x1,[1 3 7])
```





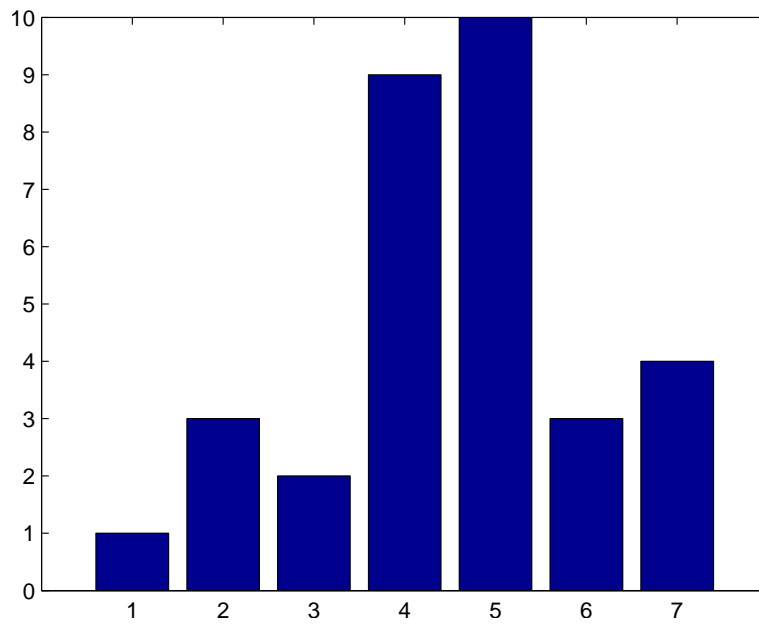
- **bar(y)** - creează un grafic cu 'bare' prin intermediul elementelor din vectorul **y**. Fiecare element din **y** indică înălțimea unei bare din cele reprezentate de **bar**.

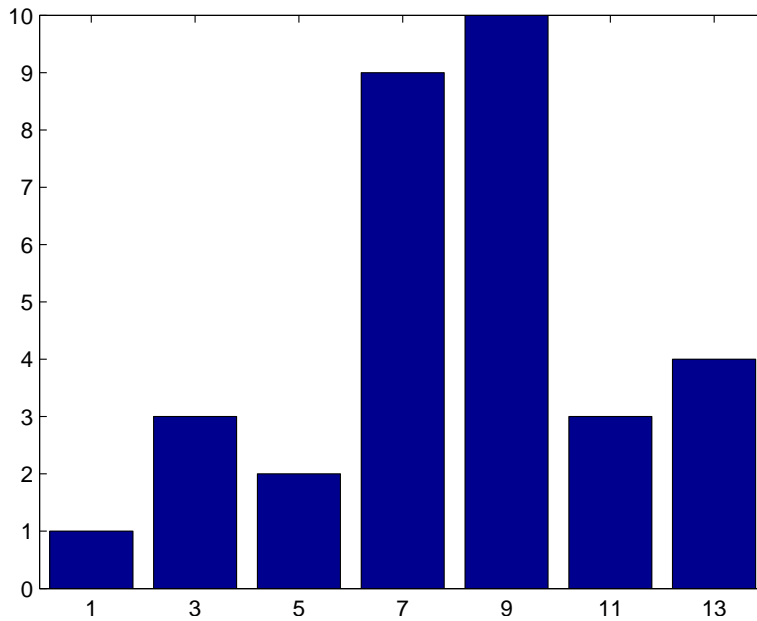
- **bar(x,y)** -poziționează 'barele' la locația specificată de vectorul de centre **x**.

Exemple:

```
y=[1 3 2 9 10 3 4];  
figure(1)  
bar(y);
```

```
figure(2)  
bar(1:2:14,y);
```





Problema I

```
function histograma(x,X,Y)

clf;
hold on;
N=length(x);
%n=ceil(1+log2(N));%->regula lui Sturges

%determinam nr de subintervale
n=ceil(2*N^(1/3)); %->regula lui Rice

f=zeros(1,n);% frecventele absolute pt fiecare subintervale

x=sort(x); %sortare vector x

d=(x(N)-x(1))/n; % lungimea unui subinterval

a=x(1)+d*(0:n); % generare diviziune echidistanta de pas d
                % a contine capetele subintervalelor
%calculez frecventele absolute pt fiecare subinterval
```

```

for i=1:N
    for j=1:n
        if a(j)<=x(i) & x(i)<=a(j+1)
            f(j)=f(j)+1;
        end
    end
end
end
p=(1/N)*f; %p contine frecventele relative
h=(1/d)*p; % probabilitatea ca un element dintr-un subinterval sa fie generat

m=x(1)+d/2+d*(0:n-1); % m=centrele pt bin-uri
bar(m,h,'histc'); %reprezentare bin-uri
plot(X,Y,'-r','LineWidth',2); % reprezentare grafica a densitatii de probabilitate
set(gca,'XTick',ceil(x(1)):1.5:ceil(x(N))); %setez markerii pt axa Ox sa mearga cu pas 1.5
end

```

Exemplu de apel:

```

clear all
x=chi2rnd(4,1,200);
X=min(x):0.001:max(x);
Y=chi2pdf(X,4);
figure(1)
histograma(x,X,Y);

clear all

x=betarnd(2,3,1,200);
X=min(x):0.01:max(x);
Y=betapdf(X,2,3);

figure(2)
histograma(x,X,Y);

```

Problema II

```

function test(N)
distributie=[round(0:30); binopdf(0:30,30,1/4)]
A=binornd(30,1/4,1,N);
hist(A,15);

```

```
set(gca,'XTick',0:30);  
end
```

Problema III

Pentru un n fixat vom genera folosind legea exponențială n numere ce reprezintă numărul de viruși depistați în fiecare zi. Se va calcula media acestor zile și se va repeta procesul de N ori.

Algoritmul va trebui testat pentru fiecare n de la 1 la 20. (for $n = 1 : 20$).

Exemplu:

```
%n fixat  
-----  
clf  
for i=1:N  
    virusi_pe_n_zile=exprnd(0.01,1,n);  
    media_n_zile(i)=mean(virusi_pe_n_zile);  
end  
hist(media_n_zile);  
pause(5)  
-----
```