Laborator 8

Funcţii utilizate:

1. **cumsum(A)** - returnează sumele cumulative folosind elementele dintr-un vector A

   cumsum(A), A matrice : rezultatul este o matrice ce contine sumele cumulative de pe fiecare coloana din A

   cumsum(A,dim) : operează pe dimensiunea specificată de **dim**

```
>>A = 1:5;
>>B = cumsum(A)
B =

1     3     6    10    15
```

```
>>A = [1 4 7; 2 5 8; 3 6 9]
>>B = cumsum(A)
B =

1     4     7
3     9    15
6    15    24
```

2. **sort(X)** - sortează vectorul X (în cazul matricilor sortarea se face pe coloane)
3. **ecdf** - **[f,x]=ecdf(y) obţine funcţia de repartiţie f a distribuţiei empirice calculată în punctele din vectorul x, folosind datele din vectorul y.**

Problema I

Generăm N numere pseudo-aleatoare conform distribuţiei date de x şi p, iar pentru acest lucru vom crea o funcţie *discret* **de parametrii x,p şi N.**

Paşi pentru crearea funcţiei:

a) ordonăm descrescător probabilităţile:

```
p=[0 sort(p,'descend')];
```

b) calculăm sumele cumulative:

```
q=cumsum(p);
```

c)găsim indicele j pentru care $p_0 + p_1 + .. + p_{j-1} < U < p_0 + p_1 + .. + p_j$:

```
a=[];
for i=1:N
      U=unifrnd(....);
j=1;
while q(j)<U
.....
end
%adaugam x(j-1) in vectorul a
```

Găsim frecvenţele cerute:

```
function urna_simulare(N)
y=discret(1:4,[0.46 0.4 0.1 0.04],N)
[f,x]=ecdf(y);
probabilitati=[0.46 0.4 0.1 0.04]

%Are loc proprietatea P(i-1<=X<i)=f(i)-f(i-1), in cazul nostru P(i-1<=X<i)=P(X=i-1)
frecvente=[f(2)-f(1) f(3)-f(2) f(4)-f(3) f(5)-f(4)]
end
```

Problema II

Generăm N numere pseudo-aleatoare conform legii exponenţiale cu valoarea medie lambda, iar pentru aceasta vom crea funcţia *expAleator*, **de parametrii lambda şi N:**

```
function   ....
a=[];
for i=1:N
      U=unifrnd(....);
%generam numarul cu ajutorul functiei de repartie corespunzatoare legii exponentiale
%y=F^(-1)(U)
%adaugam y la a
....
end
end
```

Reamintim densitatea de probabilitate a legii exponenţiale:

$$f(x) = \begin{cases} \lambda e^{-\lambda x} & x > 0 \\ 0 & x \leq 0 \end{cases}$$

iar funcţia de repartiţie este $F(x) = \int_{-\infty}^{x} f(t)dt$**. Vom determina funcţia de repartiţie:**

Cazul I: $x \leq 0 \Rightarrow F(x) = \int_{-\infty}^{x} 0dt = 0$**.**

Cazul II: $x > 0 \Rightarrow F(x) = \int_{-\infty}^{x} f(t)dt = \int_{-\infty}^{0} 0dt + \int_{0}^{x} \lambda e^{-\lambda t}dt = 0 + (-e^{\lambda t})\Bigg|_{t=0}^{t=x} = 1 - e^{\lambda x}$**.**

Deci,

$$F(x) = \begin{cases} 1 - e^{-\lambda x} & x > 0 \\ 0 & x \leq 0 \end{cases}$$

Vom calcula inversa lui F**. Mai întâi observăm că numărul** $U(i) \in [0, 1]$**, prin urmare vom calcula inversa doar pe** $[0, \infty]$**:**

$$F(x) = y \iff 1 - e^{-\lambda x} = y \iff e^{-\lambda x} = 1 - y \iff -\lambda x = \ln(1 - y) \iff x = \frac{-1}{\lambda} \ln(1 - y)$$

Inversa este

$$F(t) = \frac{-1}{\lambda} \ln(1 - t).$$

Determinăm frecvenţele cerute:

```
function print_file(N,k)
y=exp_aleator(12,N)

%distributia empirica pentru valorile care NU sunt mai mari sau egale decat k
[f,x]=ecdf(y,'censoring',y>=k)


frecventa_ceruta=f(length(x))
probabilitatea_ceruta=expcdf(k,12)
end
```