

Labor 6 - 2017

A1. Sei das Quadrat $[0, 10] \times [0, 10]$ und gegeben ist $p \in (0, 1)$.

```
clc
figure
rectangle('Position', [0,0,10,10], 'LineWidth', 3)
grid on
hold on
line([0 1], [1 1], 'LineWidth', 3, 'Color', 'red', 'LineStyle', '--')
pause(0.3)
plot([1 1], [1 2], 'LineWidth', 3, 'Color', 'red', 'LineStyle', '--')
pause(0.3)
plot([1 2], [2 2], 'LineWidth', 3, 'Color', 'red', 'LineStyle', '--')
```

Man zeichne ein zufälliges Durchqueren des Quadrates (mit einer anderen Farbe): man startet links oben (Koordinate (0,10)) und endet rechts unten (Koordinate (10,0)) und die Bewegung ist nur: waagrecht nach rechts mit Wahrscheinlichkeit p , senkrecht nach unten mit Wahrscheinlichkeit $1 - p$. Man simuliere mehrere zufällige Durchqueren des Quadrates. Insgesamt wie viele Wege gibt es das Quadrat zu durchqueren?

A2. Man generiere die ersten n Zahlen von Bell $B_1 = 1, B_2 = 2, B_3 = 5, B_4 = 15 \dots$ (siehe Bild nebenan - das Bellsche Dreieck).

Bemerkung: B_n ist die Anzahl der Partitionen einer n -elementigen Menge.

1					
1	2				
2	3	5			
5	7	10	15		
15	20	27	37	52	
52	67	87	114	151	203
203	255	322	409	523	674
877	...				

A3. (Random shuffling - zufälliges Mischen) Ohne *randperm* oder *randsample* zu benutzen, generiere man eine zufällige Permutation einer gegebenen Folge. Den Befehl *rand* kann (sollte) man benutzen.

Anwendung:

Mit Hilfe von Simulationen schätze man die Wahrscheinlichkeiten in folgender Aufgabe und man vergleiche die Ergebnisse mit den theoretischen Resultaten.

4 Jungen und 3 Mädchen gehen gemeinsam ins Kino und kaufen Karten in einer Reihe mit genau 7 Plätzen. Die Karten werden zufällig ausgeteilt. Welches ist die Wahrscheinlichkeit, dass

- a) zwei Mädchen nicht nebeneinander sitzen? b) zwei Jungen nicht nebeneinander sitzen?
- c) Jungen an jedem Ende der Reihe sitzen? d) Mädchen an jedem Ende der Reihe sitzen?

```
X=['a','b','c','d','e','f','g'] ;
n = length(X);
for i = 2:n % forward
    w = ceil(rand*i); % 1 <= w <= i % a random number = floor(rand*i)+1
    t = X(w);
    X(w) = X(i);
    X(i) = t;
end
X
%%%%%%%%%%%%%
X=['a','b','c','d','e','f','g','h'] ;
n = length(X);
for i = n:-1:2 % backward
    w = ceil(rand*i); % a random number = floor(rand*i)+1
    t = X(w);
    X(w) = X(i);
    X(i) = t;
end
X
```

Laborator 6 - 2017

A1. Fie pătratul $[0, 10] \times [0, 10]$ și fie $p \in (0, 1)$.

```
clc
figure
rectangle('Position', [0,0,10,10], 'LineWidth', 3)
grid on
hold on
line([0 1], [1 1], 'LineWidth', 3, 'Color', 'red', 'LineStyle', '--')
pause(0.3)
plot([1 1], [1 2], 'LineWidth', 3, 'Color', 'red', 'LineStyle', '--')
pause(0.3)
plot([1 2], [2 2], 'LineWidth', 3, 'Color', 'red', 'LineStyle', '--')
```

Să se simuleze o traversare aleatoare a pătratului (cu o altă culoare): se pornește stânga sus (coordonata (0,10)) și se încheie în dreapta jos (coordonata (10,0)), iar deplasarea este doar: orizontal la dreapta cu probabilitatea p , vertical în jos cu probabilitatea $1 - p$. Să se simuleze câteva traversări aleatoare ale pătratului. Câte astfel de drumuri există?

```
1
1 2
2 3 5
5 7 10 15
15 20 27 37 52
52 67 87 114 151 203
203 255 322 409 523 674 877
877 ...
```

A2. Să se genereze primele n numere Bell $B_1 = 1, B_2 = 2, B_3 = 5, B_4 = 15 \dots$ (a se vedea figura alăturată). Observație: B_n este numărul de partiții ale unei mulțimi cu n elemente.

A3. (Random shuffling) Fără a folosi *randperm* sau *randsample*, să se realizeze o permutare aleatoare a elementelor unui șir. Comanda *rand* se poate folosi!

Aplicație: Folosind simulări estimați probabilitățile cerute în următoarele probleme. Comparați rezultatul obținut cu rezultatul teoretic.

4 băieți și 3 fete merg la cinema și au cumpărat bilete într-un rand, care are exact 7 locuri. Se împart aleator biletele.

- Care este probabilitatea ca două fete să nu stea una lângă alta?
- Care este probabilitatea ca doi băieți să nu stea unul lângă altul?
- Care este probabilitatea să stea băieți la fiecare capăt al rândului?
- Care este probabilitatea să stea fete la fiecare capăt al rândului?

```
X=['a','b','c','d','e','f','g'] ;
n = length(X);
for i = 2:n % shuffle in forward direction:
    w = ceil(rand*i); % 1 <= w <= i % choose a random number = floor(rand*i)+1
    t = X(w);
    X(w) = X(i);
    X(i) = t;
end
X
%%%%%%%%%%%%%
X=['a','b','c','d','e','f','g','h'] ;
n = length(X);
for i = n:-1:2 % shuffle in backward direction:
    w = ceil(rand*i); % 1 <= w <= i % choose a random number = floor(rand*i)+1
    t = X(w);
    X(w) = X(i);
    X(i) = t;
end
X
```