

Laborator 3 – 08.11.2023

1. Simulați experimentul aruncării repetate a unei monede. Calculați numărul necesar de aruncări pentru a obține două fețe de același fel consecutive.
2. Să se arunce două zaruri până când se obțin două numere egale (o dublă). Să se afișeze de câte ori a fost nevoie să se arunce zarul.

Observație: Aruncarea unui zar este simulată prin alegerea aleatoare a unuiu din numerele 1,2,3,4,5,6. Pentru aceasta se folosește instrucțiunea sample:

sample(**x** = vectorul din care se alege aleator, **size** = câte numere se pot genera, **replace** = FALSE/TRUE - numerele generate se repetă, **prob** = **NULL** - probabilități egale sau se indică printr-un vector fiecare dintre probabilități)

3. Să se scrie o funcție care verifică dacă un număr specificat ca argument este prim sau nu.

Observație: Cum putem demonstra că este prim? Presupunem că numărul este prim și verificăm cazurile particulare; dacă este 0 sau 1, schimbăm presupunerea. Apoi căutăm un divizor în intervalul $[2, \sqrt{n}]$, parcurgând numerele din intervalul dat, iar dacă îl găsim, schimbăm presupunerea.

4. Să se afișeze toate numerele Fibonacci mai mici de 1000.

Observație:

Numerele Fibonacci sunt definite prin următoarea relație de recurență:

$$F_0 = 0, F_1 = 1, F_i = F_{i-1} + F_{i-2} \text{ pentru } i \geq 2.$$

Astfel, fiecare număr Fibonacci este suma celor două numere Fibonacci anterioare, rezultând secvența:

0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, ...

Primele 22 de numere din șir sunt:

0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610, 987, 1597, 2584, 4181, 6765, 10946^[1]

(https://ro.wikipedia.org/wiki/Num%C4%83r_Fibonacci)

5. Analizând datele din setul de date de la ultimul exercițiu de la laboratorul precedent să se construiască un grafic alegând variabilele și specificând numele variabilei pe fiecare axă, titlul graficului, etc.

Ultimul exercitiu din laboratorul anterior: a) Să se construiască un dataframe care conține doar variabilele Manufacturer, Make, Price, Passengers și Origine.