

LABORATOR #2

EX#1 Creați un fișier în Python® prin care:

- (a) să se simuleze aruncarea cu o monedă corectă i.e. $\mathbb{P}(H) = 1/2, \mathbb{P}(T) = 1/2$;
- (b) să se simuleze aruncarea cu o monedă măsluită: $\mathbb{P}(H) = 0.7, \mathbb{P}(T) = 0.3$;
- (c) să se simuleze câte N aruncări cu moneda corectă, respectiv măsluită, și să se estimeze numeric (frecvenționist), pentru fiecare caz, probabilitatea să cadă H ;
- (d) să se afișeze într-un sistem xOy graficul estimărilor obținute la (c) în funcție de numărul de aruncări $i = \overline{1, N}$.

EX#2 Creați un fișier în Python® prin care:

- (a) să se simuleze aruncarea cu un zar corect i.e. $\mathbb{P}(\{i\}) = 1/6, i = \overline{1, 6}$;
- (b) să se simuleze aruncarea cu un zar măsluit: $\mathbb{P}(\{1\}) = 0.25, \mathbb{P}(\{2\}) = 0.25, \mathbb{P}(\{3\}) = 0.2, \mathbb{P}(\{4\}) = 0.15, \mathbb{P}(\{5\}) = 0.1, \mathbb{P}(\{6\}) = 0.05$;
- (c) să se simuleze câte N aruncări cu zarul corect, respectiv măsluit, și să se estimeze numeric (frecvenționist), pentru fiecare caz, probabilitatea să se obțină fața $\{i\}$, pentru $i \in \{1, 2, 3, 4, 5, 6\}$ fixat.

EX#3 Creați un fișier în Python® prin care:

- (a) să se estimeze numeric (frecvenționist) probabilitatea ca din 10 de aruncări cu o monedă corectă să pice cel puțin o secvență consecutivă de (H, H, H, H) sau (T, T, T, T) ;
- (b) să se estimeze numeric (frecvenționist) probabilitatea ca din 20 de aruncări cu o monedă corectă să pice cel puțin o secvență consecutivă de (H, H, H, H) sau (T, T, T, T) ;
- (c) să se estimeze numeric (frecvenționist) probabilitatea ca din 100 de aruncări cu o monedă corectă să pice cel puțin o secvență consecutivă de $(H, H, H, H, H, H, H, H, H)$ sau $(T, T, T, T, T, T, T, T, T)$.

EX#4 Un joc pentru 2 persoane presupune 3 zaruri: unul roșu cu fețele $\{1, 4, 4, 4, 4, 4\}$, unul verde cu fețele $\{3, 3, 3, 3, 3, 6\}$ și unul negru cu fețele $\{2, 2, 2, 5, 5, 5\}$. Un jucător alege un zar dintre cele 3, iar al doilea alege unul dintre cele 2 zaruri rămase. Ambii jucători aruncă cu zarurile alese și câștigă cel care obține numărul mai mare. Creați un fișier în Python® prin care:

- (a) să se simuleze jocul de mai sus;
- (b) să se estimeze numeric (frecvenționist) probabilitatea ca cel care alege primul să câștige, pentru fiecare alegere posibilă.

Indicații Python®: `numpy`, `numpy.random.rand`, `matplotlib`, `matplotlib.pyplot`