

Probabilidade Computacional – Exercício 1

1) Em simulação estocástica, nos interessa ter uma sequência de números pseudoaleatórios distribuídos de forma uniforme entre 0 e 1.

Um possível algoritmo que pode ser utilizado para gerar a sequência desejada é o LCG e cuja relação de recorrência é apresentada a seguir:

$$x_k = (a \times x_{k-1} + c) \bmod M$$

$$u_k = x_k / M$$

Outro possível método para geração de números pseudoaleatórios, chamado aqui de algoritmo LM, toma como base o lançamento de uma moeda comum que pode gerar dois eventos, cara ou coroa, cada um com probabilidade 0,5. Transformando estes eventos em números binários sendo cara = 0 e coroa = 1, podemos, assim, gerar números decimais a partir de números binários segundo a formulação abaixo usando os lançamentos de uma moeda para definir os coeficientes a_i :

$$\text{Número} = a_n 2^{(n-1)} + \dots + a_4 2^3 + a_3 2^2 + a_2 2^1 + a_1 2^0$$

Por exemplo, se $n = 5$, o que equivale a fazer 5 lançamentos de uma moeda, podemos criar 32 números decimais de 0 a $(2^5 - 1) = 31$. Supondo um número binário de 5 dígitos $(a_5 a_4 a_3 a_2 a_1) = 10011$, obtido a partir de 5 lançamentos de uma moeda, obtemos o número decimal:

$$\text{Número} = 1x2^4 + 0x2^3 + 0x2^2 + 1x2^1 + 1x2^0$$

$$\text{Número} = 1x16 + 0x8 + 0x4 + 1x2 + 1x1 = 19$$

Se dividirmos o número obtido 19 pelo maior número possível 31, geramos o valor 0,6129 que consequentemente está no intervalo $[0, 1]$. Assim, a partir deste processo, é possível gerar números pseudoaleatórios com distribuição uniforme no intervalo $[0, 1]$.

Um algoritmo mais robusto que os apresentados acima é o Mersenne Twister que fornece uma geração rápida com alta qualidade de aleatoriedade. A partir do que foi exposto e usando os conceitos de simulação, faça:

a) Construa um código baseado no LCG para realizar uma simulação com 10.000 lançamentos simultâneos de duas moedas, ambas honestas, e apresentar em quantos lançamentos o resultado obtido foi **cara** e **coroa**. O LCG deve ser utilizado para tratar o lançamento de ambas as moedas.

b) Construa um código baseado no LCG e no Mersenne Twister para realizar uma simulação com 10.000 lançamentos simultâneos de uma moeda e de um dado (8 faces), ambos honestos, e apresentar em quantos lançamentos o resultado obtido foi simultaneamente **coroa** e **face 5**. O LCG deve ser utilizado para tratar o lançamento da moeda e o Mersenne Twister para tratar o lançamento do dado.

c) Construa um código baseado no LCG e no LM (com lançamento de **10** moedas) para realizar uma simulação com 10.000 lançamentos simultâneos de uma moeda honesta e de um dado (8 faces) viciado (probabilidade de obter a **face 3** é zero e é a mesma para as demais faces) e apresentar em quantos lançamentos o resultado obtido foi simultaneamente **coroa** e **face 8**. O LCG deve ser utilizado para tratar o lançamento da moeda e o LM para tratar o lançamento do dado.

d) Construa um código baseado no Mersenne Twister e no LM (com lançamento de **10** moedas) para realizar uma simulação com 10.000 lançamentos simultâneos de uma moeda viciada (probabilidade de tirar **cara** é **0,45**) e de um dado (**8** faces) honesto, e apresentar a probabilidade de se obter pelo menos um dos resultados: simultaneamente **cara** e **face 1**; simultaneamente **cara** e **face 4**; simultaneamente **coroa** e **face 7**. O Mersenne Twister deve ser utilizado para tratar o lançamento da moeda e o LM para tratar o lançamento do dado.