Trabalho Individual sobre Números (Pseuso)-Aleatórios

Não é simples a geração de números aleatórios para uso em sistemas de segurança computacional. Normalmente, estamos interessados em números grandes, da ordem de grandeza de centenas de dígitos. No Brasil, por exemplo, para assinar documentos eletrônicos, você vai precisar ter chaves criptográficas geradas a partir de números aleatórios de 2048 bits.

Neste trabalho individual, vamos explorar técnicas para se gerar números pseudo-aleatórios.

1) Quanto ao Entregável

Você deve entregar no Moodle um ÚNICO arquivo PDF com os seguinte requisitos:

A. Uma seção para explicar os algoritmos de Números Aleatórios escolhidos;
B. Os códigos devem estar no PDF (incluído no PDF) e devidamente documentados (**comentados**);
C. Referências a cada um dos algoritmos.

LEMBRE-SE: Entregue um **único documento PDF**, contendo o relatório desse trabalho individual (incluindo os códigos dos programas, tabelas, saídas, …)

2) Gerar Números Pseudo-aleatórios

Você deve escolher **três (3)** dos seguintes algoritmos geradores de números pseudo-aleatórios e implementá-lo em Python (*Ou qualquer outra linguagem de programação, desde que justifique seu uso no relatório*). Você pode, opcionalmente, propor o uso de algoritmos diferentes, mas deve justificar a escolha.

- Blum Blum Shub
- Complementary-multiply-with-carry
- Inversive congruential generator
- ISAAC (cipher)
- Lagged Fibonacci generator
- Linear congruential generator
- Linear feedback shift register
- Maximal periodic reciprocals
- Mersenne twister
- Multiply-with-carry
- Naor-Reingold Pseudorandom Function
- Park–Miller random number generator
- Well Equidistributed Long-period Linear
- Xorshift

Você deve executar cada um dos algoritmos escolhidos para:

- Gerar números pseudo-aleatórios grandes. Entende-se como grande, números de até 4096 bits. Experimente gerar números da seguintes ordens de grandeza: 40, 56, 80, 128, 168, 224, 256, 512, 1024, 2048, 4096 bits binários;
    - Caso não seja possível gerar números aleatórios de um determinado tamanho usando um dos algoritmos, você deve justificar;
    - Monte uma tabela explicitando o algoritmo, o tamanho do número e o tempo necessário para gerá-lo;
        - Exemplo:
        -

Algoritmo	Tamanho do Número	Tempo para gerar
Blum blum Shib	40	10 micro seg
xxx	56	

- O código implementado deve estar documentado. Dá-se preferência para documentar no próprio corpo do programa, na forma de comentários;
- Compare os algoritmos escolhidos;
- Faça uma análise de complexidade dos algoritmos.
- Como medir se o número é de fato aleatório?
    - Procure na literatura formas de se avaliar e aplique em suas sequências.