Universidade Federal de Uberlândia

Modelagem e Simulação

Equipe: Bruno Borges

Jefferson Freitas Oliveira

Marilia Leal

Vinicius Scavoni

Uberlândia – 2017

Exercise 2.1.1 For the tiny Lehmer generator defined by g(x) = ax mod 127, find all the full-period multipliers.

(a) How many are there?
Existem 36 full-period multipliers quando m = 127
(b) What is the smallest multiplier?
3 é o menor full-period multiplier quando m = 127

Tela de execução do programa:

```
Todos os full-period multipliers são:
3 116 109 92 86 12 83 112 55 114 48 78 67 93 106 65 58 14 118 46 43 6 56 91 57 45 39 97 110 101 53 96 29 7 23 85


Há 36 full-period multipliers quando m = 127
3 é o menor full-period multiplier quando m = 127
```

Exercise 2.1.6 In ANSI C an int is guaranteed to hold all integer values between −(2^15 − 1) and 2^15 − 1 inclusive.

(a) What is the largest prime modulus in this range?
O maior primo no intervalo dado é 32749.
(b) b) How many corresponding full-period multipliers are there and what is the smallest one?
Há 10912 full-period multipliers quando m = 32749 e para esta entrada 2 é o menor full-period multiplier.

Exercise 2.1.8

(a) Evaluate 7 i mod 13 and 11 i mod 13 for i = 1, 5, 7, 11.

```
For a full-period 7 and m modulus 13 we have:

The a full-period is 7 for i equals 1
The a full-period is 11 for i equals 5
The a full-period is 6 for i equals 7
The a full-period is 2 for i equals 11


For a full-period 11 and m modulus 13 we have:

The a full-period is 11 for i equals 1
The a full-period is 7 for i equals 5
The a full-period is 2 for i equals 7
The a full-period is 6 for i equals 11

Process returned 0 (0x0)   execution time : 0.001 s
Press ENTER to continue.
```

(b) How does this relate to Example 2.1.5?

Todos os “a” full-period da questão, que são dados e os que são gerados, são iguais aos do exemplo. Isso acontece porque estamos tratando do mesmo ”m” e de acordo com o teorema 2.1.4 dado um a full-period de modulo m primo encontramos todos os outros full-period possíveis através da fórmula.

Exercise 2.1.9

(a) Verify that the list of five full-period multipliers in Example 2.1.6 is correct.

Para verificar os resultados do exemplo 2.1.6, o código foi implementado para retornar os 10 primeiros full-period multiplier, com m=2^31-1.

Sabendo que se “i” está no intervalo entre [1,m-1], se “i” e “m-1”, forem relativamente primos, então (a^i) mod m é um full periodo multiplier.

```
7^1 mod 2147483647 = 7
7^5 mod 2147483647 = 16807
7^13 mod 2147483647 = 252246292
7^17 mod 2147483647 = 52958638
7^19 mod 2147483647 = 447489615
```

Os resultados obtidos, foram compatíveis com o do exemplo.

(b) What are the next five elements in this list?

Os 5 próximos full-period multiplier encontrados foram:

```
7^23 mod 2147483647 = 680742115
7^25 mod 2147483647 = 1144108930
7^29 mod 2147483647 = 373956417
7^37 mod 2147483647 = 655382362
7^41 mod 2147483647 = 1615021558
```

Exercise 2.1.11 For the first few prime moduli, this table lists the number of full-period multipliers and the smallest full-period multiplier. Add the next 10 rows to this table.

Módulo primo M	Número de full-period multipliers	Menor full-period multiplier
17	8	3
19	6	2
23	10	5
29	12	2
31	8	3
37	12	2
41	16	6
43	12	3
47	22	5
53	24	2

Exercise 2.2.9 You have been hired as a consultant by XYZ Inc to assess the market potential of a relatively inexpensive hardware random number generator they may develop for high-speed scientific computing applications. List all the technical

reasons you can
think of to convince them this is a bad idea.

Construir um gerador físico de números aleatórios é um desperdício de recursos e de tempo, visto que um gerador de números aleatórios utilizando algoritmos é o único que consegue garantir os 5 critérios de geração aleatória sendo eles:

Aleatoriedade: Capaz de produzir resultados satisfatoriamente aleatórios que podem ser provados por testes de estatística.

Controlabilidade: Se for desejável é possível repetir os resultados utilizando os mesmos valores de entrada.

Portabilidade: Capaz de reproduzir os mesmos resultados em sistemas computacionais diferentes.

Eficiência: Rápido mesmo com o mínimo de recursos computacionais.

Documentação: Teoricamente analisado e com grande quantidade de testes.

Exercise 2.2.11 Let m be the largest prime modulus less than or equal to 2^15 − 1 (see Exercise 2.1.6).

(a) Compute all the corresponding modulus-compatible full-period multipliers.

Resultados no código

(b) Comment on how this result relates to random number generation on systems that support 16-bit integer arithmetic only.

Esse resultado possui um melhor desempenho pois com o método antigo g(x)= a^x mod m, conforme os valores de x fossem crescendo, a^x se tornava um número muito grande que não conseguia ser computado dependendo das configurações do sistema. Com essa fórmula a multiplicação de ‘a’ é feita após a operação de mod: x = a* (x % q), onde q <r e q=m/a e r=m%a.

Exercise 2.2.15 Determine whether the multipliers associated with m = 2 31 − 1 given
by Fishman (2001): a = 630 360 016, a = 742 938 285 a = 950 706 376, a = 1 226 874 159,
a = 62 089 911, and a = 1 343 714 438 are modulus-compatible.

```
630360016 isn't module compatible with 2147483647
742938285 isn't module compatible with 2147483647
950706376 isn't module compatible with 2147483647
1226874159 isn't module compatible with 2147483647
62089911 isn't module compatible with 2147483647
1343714438 isn't module compatible with 2147483647

Process returned 0 (0x0)   execution time : 0.002 s
Press ENTER to continue.
```

Exercise 2.3.6 According to slides number seven and eight from section 2.3, example 2.3.6, construct a graph similar to slide eight but Pr(X=9).