Instituto Nacional de Matemática Pura e Aplicada

Aluno: Hallison da Paz

Curso de Algoritmos
Trabalho 0 (Aquecimento)

Rio de Janeiro, agosto de 2015.

1 Pesquise sobre o Algoritmo de Euclides

O algoritmo de Euclides, utilizado para determinar o máximo divisor comum (MDC) entre dois números naturais, baseia-se sobre o fato de que o MDC entre dois números naturais quaisquer é igual ao MDC entre o menor destes números e a diferença entre o maior e o menor número. Isto é:

- Sejam **a** e **b** dois números naturais quaisquer. Sem perda de generalidade, suponha $a \geq b$. Se **k** é o MDC entre **a** e **b**, então **k** também é o MDC entre **b** e **a-b**.

Desta forma, no cálculo do MDC entre a e b, podemos substituir o maior deles por max(b, a-b) e o menor por min(b, a-b). O algoritmo para no momento em que o menor número iguala-se a zero, devolvendo o resultado da subtração imediatamente anterior. A convergência do algoritmo é muito mais rápida se utilizarmos a aritmética modular, substituindo sucessivas subtrações do mesmo número pelo resto da divisão entre os dois; implementaremos desta forma.

2 Implemente o algoritmo de Euclides

Este algoritmo encontra-se implementado no arquivo *WarmUp.py* (função mdc). O algoritmo foi implementado na linguagem de programação Python versão 3.4 [1] e faz uso de recursos específicos de Python 3, como *function annotations,* não sendo retrocompatível com versões 2.x desta linguagem.

3 Meça o número de passos feitos pelo algoritmo de Euclides para calcular o o máximo divisor comum de dois números naturais a e b, para 1 ≤ a,b ≤ N. Como o número máximo varia em função de N? Como o número médio varia em função de N?

O experimento foi conduzido com valores de N entre 1 e 1000 inclusive. Foram tomados todos os pares de números **a** e **b** nestes intervalos, de modo que 1 ≤ a,b ≤ N. A figura 1 ilustra o resultado obtido durante o experimento. A curva *max_value* representa o comportamento do número máximo de passos executados pelo algoritmo; a curva *avg_value*, por sua vez, refere-se ao número médio de passos verificados. Adicionalmente, incluiu-se a curva min_value para mostrar também o comportamento do número mínimo de passos, que permaneceu constante em 1.

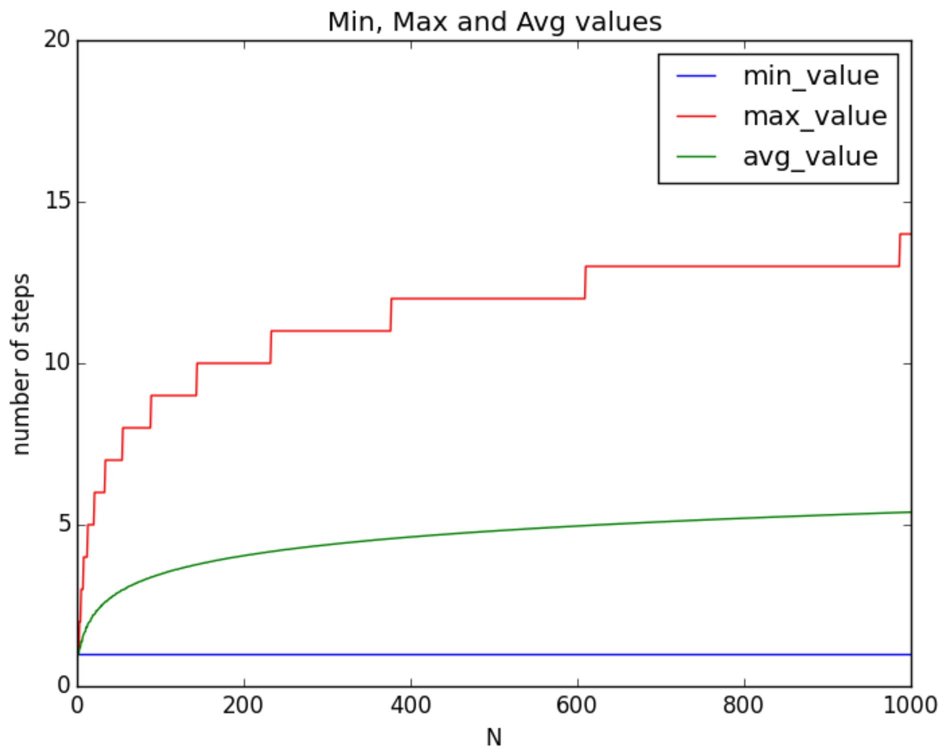


Figure 1: Número de passos feitos pelo algoritmo

Analisando a figura 1, percebemos que o número máximo de passos tem um comportamento monótono não decrescente. A curva de máximo apresenta um aspecto similar ao de uma "escada", isto é há pontos de transição em que a função cresce e trechos em que ela permanece constante.

Com relação ao número médio de passos, por estamos lidando com uma média, permitimos valores fracionários e verificamos uma curva monótona crescente, de aspecto côncavo, isto é, sua taxa de crescimento reduz à medida que o valor de N aumenta.

4 Pesquise sobre o desempenho do algoritmo de Euclides.

O desempenho do algoritmo de Euclides depende dos valores de entrada e da relação que esses valores possuem entre si. Por exemplo, sejam a,b com $a \geq b$ e $1 \leq a, b \leq N$. O número de passos necessários para computar o MDC entre a e b com este algoritmo pode ser analisado em três situações:

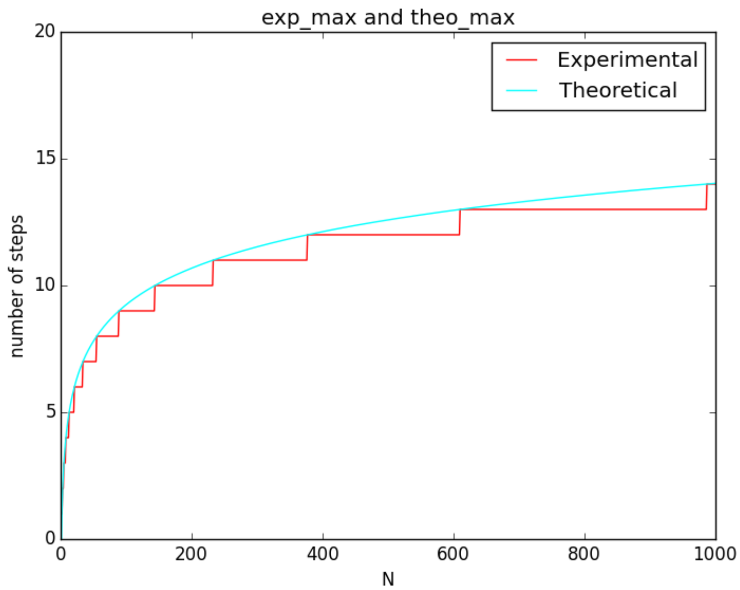
1. Quando **b** é igual a 1ou é o próprio MDC entre eles, o algoritmo executa um único passo. Este é o *melhor caso* possível.

2. Gabriel Lamé provou em 1884 que $passos \leq \frac{\ln n}{\ln \phi} + \frac{\ln \sqrt{5}}{\ln \phi} - 2$, em que ϕ é a razão áurea [2], ou numericamente: $passos \leq 4.785 \log_{10} n - 0,3277$. Neste caso, n corresponde ao menor dos números entre **a** e **b**. Fazendo a e b variarem entre todos os números naturais possíveis entre 1 e N, podemos substituir n por N para obter uma curva que represente a variação do número máximo de passos com o valor de N e se quisermos a quantidade exata de passos para um dado N, devemos tomar o piso do resultado (visto que são quantidades inteiras).

3. O número médio de passos, quando computado em relação à média de passos executados ao escolhermos **a** e **b** aleatoriamente com distribuição uniforme é dado por: $\frac{12}{\pi^2} \ln 2 \ln N + 0.06$ [4]

5 Compare as previsões teóricas com os seus resultados experimentais.

A figura 2 ilustra a curva de crescimento da previsão teórica para o número máximo de passos juntamente com a curva experimental. Nesta figura, é possível verificar que o resultado experimental está perfeitamente compatível com o resultado teórico. A curva teórica, dada por $passos \leq 4.785 \log_{10} n - 0,3277$, atua como uma cota superior para o número máximo de passos, coincidindo com os experimento exatamente nos ponto de transição (quando o valor experimental cresce). Ainda na figura, é possível ver que se aproximarmos o valor da curva teórica para o maior inteiro menor que este valor, ou seja: tomarmos $passo = \llcorner 4.785 \log_{10} n - 0,3277 \lrcorner$, teremos as duas curvas coincidindo, o que é ilustrado na figura 3.



s

Figure 2: Curvas de máximo experimental e teórica

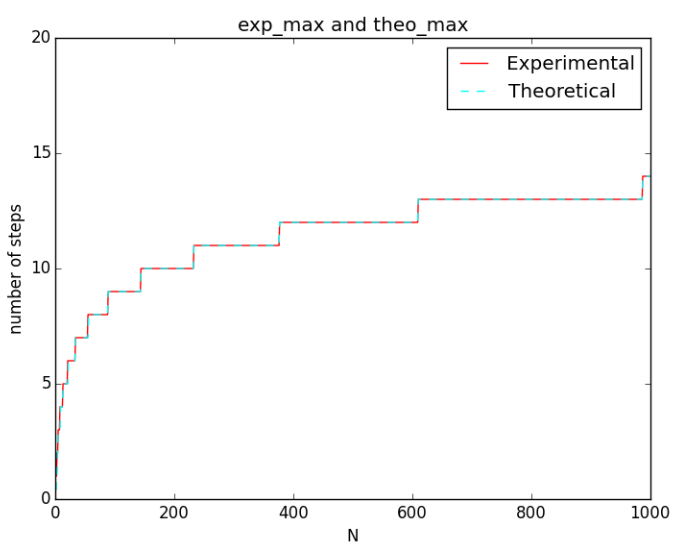


Figure 3: Curva de máximo experimental e piso do valor teórico

A figura 4 ilustra, de maneira similar, a curva esperada pela previsão teórica para o número médio de passos do algoritmo e a curva obtida experimentalmente. É possível perceber que ambas as curvas são monótonas crescentes e côncavas. O resultado experimental apresentou valores um pouco abaixo do resultado teórico, pois a previsão teórica é calculada tomando-se **a** e **b** aleatoriamente, enquanto nosso experimento tomou todos os pares possíveis entre 1 e N, inclusive, sem repetição.

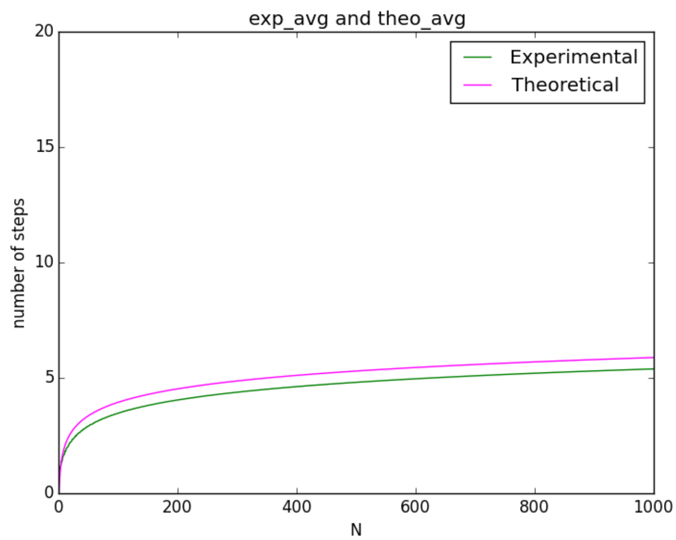


Figure 4: Curvas de valor médio experimental e teórica

Adicionalmente, a figura 5 ilustra o resultado do experimento conduzido a partir de amostragens aleatórias, uniformemente distribuídas, dos valores a e b dentro de um limite N estabelecido. O valor de N variou entre 1 e 1000, inclusive, e para cada N foram realizadas 1 milhão de amostras. Neste caso, o procedimento de cálculo da média foi realizado da mesma forma que o cálculo da previsão teórica [4] que resulta em $\frac{12}{\pi^2} \ln 2 \ln N + 0.06$, fazendo com que as curvas sejam muito próximas.

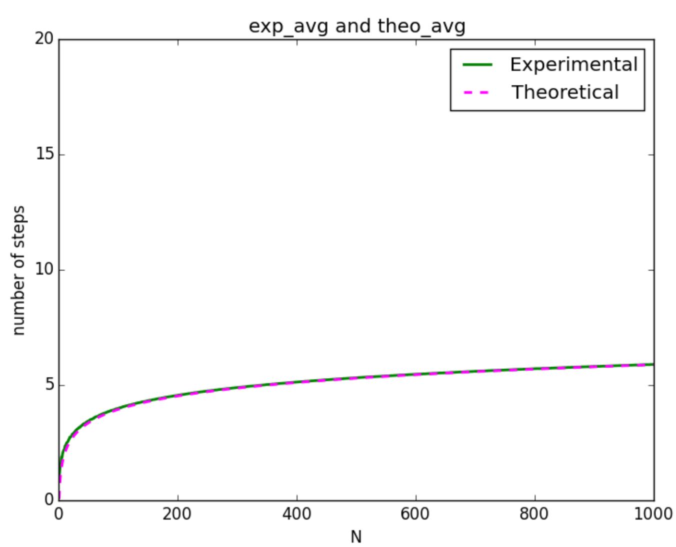


Figure 5: Curvas de média por amostragem aleatória e teórica

6 Extra: análise do pior caso

Segundo o Teorema de Lamé [2], se o algoritmo de Euclides executa P passos para calcular o MDC entre dois números **a** e **b**, com $a \geq b \geq 1$, então os menores números que atendem à esta condição são $a = F_{p+2}$ e $b = F_{p+1}$, em que $\{F_k\}$é a sequência de Fibonacci. Portanto, teoricamente, quando os pares selecionados são ambos números da sequência de Fibonacci, ocorre o pior caso computacional do algoritmo de Euclides. Podemos verificar isto experimentalmente requisitando a impressão dos valores de **a** e **b** para um arquivo, assim como o número máximo de passos para cada valor de N. A tabela 1 apresenta o resultado obtido para os pares que produziram o "pior caso", isto é, o maior número de passos para N entre 1 e 1000 inclusive.

N	Passos	a	b		N	Passos	a	b
1	1	1	1		55	8	55	34
3	2	3	2		89	9	89	55
5	3	5	3		144	10	144	89
8	4	8	5		233	11	233	144
13	5	13	8		377	12	377	233
21	6	21	13		610	13	610	377
34	7	34	21		987	14	987	610

Table 1: Pares que produziram os piores casos entre 1 e 1000

Para os valores de N entre os valores representados na tabela, os pares de número se repetiram até o próximo elemento da sequência, como esperado. O arquivo *worst_ case.csv* apresenta o resultado completo do experimento, a tabela 1 foi obtida pela filtragem deste arquivo com a função *filter_ worst_ cases* em *WarmUp.py*.

References

[1] Python; [internet] Disponível em: <https://www.python.org/> [acesso em 12 de agosto de 2015]

[2] Lamé's Theorem - the Very First Application of Fibonacci Numbers. [internet] Disponível em: <http://www.cut-the-knot.org/blue/LamesTheorem.shtml> [acesso em 13 de agosto de 2015]

[3] What is the time complexity of Euclid's Algorithm (Upper bound,Lower Bound and Average)?: [internet] Disponível em: <http://math.stackexchange.com/questions/258596/what-is-the-time-complexity-of-euclids-algorithm-upper-bound-lower-bound-and-a> [acesso em 11 de julho de 2015]

[4] Euclidean algorithm, Algorithmic_efficiency. [internet] Disponível em: <https://en.wikipedia.org/wiki/Euclidean_algorithm
[acesso em 12 de agosto de 2015]