

Último Primo Da Faixa

Mário Leite

...

Como já escrevi em publicações anteriores: os números primos são fascinantes, rebeldes, misteriosos, intrigantes e desafiadores; eles não seguem uma regra geral de criação tal como acontece com os números compostos. Muitos tentaram, e ainda tentam compreendê-los, na esperança de jogar mais luz sobre este assunto. Eratóstenes, Mersenne, Fermat e Euler são os que mais estudaram os primos, dando muitas contribuições para a Teoria dos Números; entretanto, uma Lei de Formação da sequência desses números ainda não foi publicada de maneira clara e objetiva: é o calo dos matemáticos. Eles são tão imponentes que, qualquer número par, a partir de 4 pode ser formado pela soma de dois primos. Existe até uma dúvida a respeito deles: são infinitos ou não!? O que se sabe hoje é que o último primo detectado (até 31/12/2018) tem mais de VINTE E QUATRO MILHÕES de dígitos; um número tão grande, que é muito maior que o número total de átomos que existe em todo o Universo, que é de “apenas” 10 elevado a 80 átomos (o número 1 seguido de oitenta zeros...). Assim, cansado de pesquisar, no Google, o último número primo de uma determinada faixa, resolvi criar um algoritmo que mostrasse esse número, pois, até agora só consegui encontrar tabelas de primos na faixa de 2 a 1 quatrilhão (primo com até 16 dígitos); a partir daí não encontrei (ainda) nenhuma tabela que mostre o último primo de uma faixa.

O programa “**GeraUltimoPrimo**”, mostrado abaixo (codificado em Python) é uma solução para esse problema; mostra, inclusive, o tempo de processamento para cada faixa pesquisada. É claro que esses tempos dependem muito do sistema computacional empregado: linguagem, memória RAM, espaço livre em disco, memória *cache* e outros fatores; mas, em princípio, num PC com 4 GB de RAM e 2 Ghz (no qual testei o programa) os resultados são bem confiáveis. Entretanto, a partir de 1 septilhão (número com 25 dígitos) desisti de processar, pois o meu PC ficou rodando mais 24 horas seguidas e não terminou; então resolvi abortar o processamento.

Antes de terminar, gostaria de pedir desculpas aos programadores que codificam em Python por ter “inventado” terminadores para as estruturas de controle; mas, a verdade é que eu não consigo entender o porquê dessa falta no Python (com todo o respeito ao Guido van Rossum). Sinceramente, não consigo escrever linhas de código dentro de uma estrutura de controle que não tenha uma instrução que indique o final dela e, ainda mais, impondo uma “ditadura” de indentação, como acontece nessa linguagem; por isto, resolvi criar os meus próprios terminadores. E se algum amigo, de qualquer grupo de programação, conseguir melhorar o algoritmo, de maneira que possa detectar, com um tempo razoável, os últimos primos de faixas limites a partir 1 septilhão, eu agradeceria a ajuda!

Para adquirir o *pdf/book* de meus livros sobre programação, entre em contato:

marleite@gmail com

```
'''  
Programa "GeraUltimoPrimo"  
Gera o último primo num limite dado  
Em Python 3.7  
Autor: Mário Leite  
// -----  
'''  
  
endif = "endif"  
endfor = "endfor"  
endwhile = "endwhile"  
import time  
import math  
Lim = 0  
while((Lim<3) or (Lim>10000000000000000000000)): #limite de 1 sextilhão  
    Lim = int(input("Digite o limite máximo de pesquisa: "))  
endwhile  
maxLim = Lim  
inicio = time.time() #liga o cronômetro  
if(Lim % 2 == 0):  
    Lim = Lim - 1 #garante um valor ímpar  
endif  
print()  
UltPri = Lim #considera como primo o limite da faixa  
while(True):  
    if(UltPri % 5 !=0):  
        IntRaiz = math.sqrt(UltPri)  
        IntRaiz = int(IntRaiz)  
        TemDiv = False  
        IntRaiz = IntRaiz + 1  
        for k in range(2,(IntRaiz+1)): #faz as divisões  
            RDiv = (UltPri % k)  
            if(RDiv==0):  
                TemDiv = True  
                break  
            endif  
        endfor  
        if(TemDiv==False): #se não tem divisores ==> é primo  
            print(f'Último primo no intervalo [2-{maxLim}]: {UltPri}')  
            break #sai do loop externo (já encontrou o último primo)  
        else:  
            UltPri = UltPri - 2 #considera o número ímpar anterior  
        endif  
    else:  
        UltPri = UltPri - 2  
    endif  
endwhile #fim do loop para detectar o último primo procurado  
print()  
fim = time.time() #desliga o cronômetro  
tempo = fim - inicio  
tempo = int(tempo*100+0.50)/100 #calcula com duas decimais  
print(f'Tempo gasto no processamento: {tempo}')
```

#FimPrograma-----

Limite	Último Primo	Tempo
10	7	0.00 s
50	47	0.00 s
100	97	0.00 s
150	149	0.00 s
1.000	997	0.00 s
1.500	1.499	0.00 s
10.000	9.973	0.00 s
10.500	10.499	0.00 s
100.000	99.991	0.00 s
150.000	149.993	0.00 s
1.000.000	999.983	0.00 s
1.500.000	1.499.977	0.00 s
10.000.000	9.999.991	0.00 s
10.500.000	10499.981	0.00 s
100.000.000	99.999.989	0.00 s
100.500.000	100.499.999	0.00 s
1.000.000.000	999.999.937	0.00 s
1.500.000.000	1.499.999.957	0.02 s
10.000.000.000	9.999.999.967	0.02 s
10.500.000.000	10.499.999.989	0.02 s
100.000.000.000	99.999.999.977	0.06 s
100.500.000.000	100.499.999.981	0.06 s
1.000.000.000.000	999.999.999.989	0.16 s
1.500.000.000.000	1.499.999.99.9987	0.20 s
1.000.000.000.000.000	999.999.999.999.989	5.64 s
1.500.000.000.000.000	1.499.999.999.999.939	6.16 s
1.000.000.000.000.000.000	999.999.999.999.999.989	166.01 s = 2min46s
1.500.000.000.000.000.000	1.499.999.999.999.999.941	217.59 s = 3min38s
1.000.000.000.000.000.000.000	999.999.999.999.999.999.899	9892.01 s = 2h44min52s
1.000.000.000.000.000.000.000.000	????????????????????????	???????? s

Tabela de tempos de processamento para detectar o último primo de uma faixa