

## Curiosidades Sobre Número Primos

Mário Leite

...

Os números primos são fascinantes, rebeldes, misteriosos, e continuam sendo o calo de muitos matemáticos que tentam “domesticá-los” mas, em vão; pois esses números não seguem uma regra clara de criação tal como os números compostos. Muitos tentaram, e ainda tentam compreendê-los, na esperança de jogar mais luz sobre o 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. Uma das linhas de pesquisa sobre a qual eu li, diz respeito à distância (intervalo) que separa um número primo de seu consecutivo: a diferença entre eles. A partir do número 3 essa diferença é sempre um valor par do tipo  $2k$  ( $k=1,2,3,\dots,n$ ); ou seja, todas essas diferenças são múltiplos de 2; e isto, na minha modestíssima opinião, pode colaborar com alguma teoria para esclarecer mais sobre esse tipo de número, e até reacender a polêmica se eles são finitos ou infinitos. Por outro lado, é preciso esclarecer quando se fala em “números primos grandes”, exigidos na criptografia de chaves assimétricas do método RSA, por exemplo. Talvez esta expressão pode não traduzir bem a grandeza desses números. Mas, pense no tamanho do mais recente buraco negro descoberto por cientistas alemães: o Abell 85 na galáxia Holm 15A, que fica a 700.000.000 de anos-luz da Terra ( $700.000.000 \times 9.461.000.000.000 = 6.622.700.000.000.000.000$ ); que dá mais de seis sextilhões de quilômetros distante de nós! E, ainda sobre esse buraco negro, com diâmetro de 236.000.000.000 km: ele tem uma massa 40.000.000.000 de vezes maior que a massa do Sol; e, considerando a massa do Sol igual a 1.989.000.000.000.000.000.000.000 ton, encontraremos um valor de  $40.000.000.000 \times 1.989.000.000.000.000.000.000.000$ , cujo resultado final é de 79.560.000.000.000.000.000.000.000.000.000 ton (mais de 79 undecilhões de toneladas)!

E se esse último número de “apenas” trinta e oito dígitos, impressiona, imagine o tamanho do maior número primo encontrado (até Dez/2018) que tem mais de VINTE E QUATRO MILHÕES de dígitos; um número tão grande, que é muito maior que o número de átomos de todo o Universo, que é cerca de “apenas” 10 elevado a 80 átomos!

Assim, diante desta, entre muitas outras curiosidades dos números primos, resolvi criar um programa para mostrar alguns dados: a diferença entre dois primos consecutivos em dois limites definidos, a maior diferença encontrada, os primos desse intervalo e o total deles. O programa "**MostraDifPrimos**" foi codificado em Python devido ao fato de ser uma excelente linguagem de programação, com um grande poder de cálculo e muito rápida. Rodando o programa para o intervalo de 1 até 1.000.000.000, em cujo resultado pode ser notado que o último primo desta faixa é 999.999.937, e a maior diferença entre dois primos consecutivos é 282 (entre 436.273.291 e 436.273.009). Também é calculado o tempo de processamento do programa, para verificar identificar todos os números primos nesse intervalo, que foi de cerca de **159.312** segundos = **44h15min12s**, com um PC de 4Gb RAM e *clock* de 2 Ghz. Mas, o que me chamou a atenção foi que à medida que o intervalo aumenta a maior diferença também aumenta, mas, não necessariamente, entre os dois últimos pares de números; muito interessante!

E, para terminar: observando as dezenas sorteadas na Mega Sena, pode ser notado que sempre aparecem números primos; confira o resultado do concurso 2141 de 10/04/2019, com cinco números primos, entre as seis dezenas sorteadas: **10 11 17 19 37 41**, repetindo o que já havia acontecido no concurso 2124 de 13/02/19.

Tem alguma coisa misteriosa com esse tipo de número, não é?!!

---

```

'''
Programa "MostraDifPrimos"
//Mostra os primos num determinado intervalo e as diferenças entre eles.
//Autor: Mário Leite
//-----
'''
endif = "endif"
endwhile = "endwhile"
endfor = "endfor"
import math
import time
Lim1 = 1
Lim2 = 0
lstPrimos = [] #inicializa vetor de primos
Cond = (Lim1>Lim2) or ((Lim1<1) or (Lim2<2))
while(Cond):
    Lim1 = int(input("Entre com o limite inferior: "))
    Lim2 = int(input("Entre com o limite superior: "))
    Cond = (Lim1>Lim2) or (Lim1<1) or (Lim2<2)
endwhile
inicio = time.time() #liga o cronômetro
Cont = -1
for j in range(Lim1, (Lim2+1)):
    PriPrx = j + 1 #pega o primeiro número após j
    IntRaiz = int(math.sqrt(PriPrx))
    TemDiv = False
    for k in range(2, (IntRaiz+1)):
        #faz as divisões
        RDiv = (PriPrx % k)
        if(RDiv == 0):
            TemDiv = True
            break #abandona incondicionalmente o loop (não é primo)
        endif
    endfor
    if(TemDiv==False): #checa se é primo
        Cont = Cont + 1
        #lstPrimos[Cont] = PriPrx
        lstPrimos.append(PriPrx)
        print(PriPrx)
    endif
endfor #fim do loop para detectar todos os primos no intervalo
print(f'Quantidade de primos encontrada: {Cont}')
print('')
MaiorDif = 0
#Pesquisa e mostra os intervalos entre dois primos consecutivos
print("Diferenças entre primos consecutivos:")
j = 2
while(j<=Cont):
    DifIni = lstPrimos[j-1]
    DifFim = lstPrimos[j]
    Dif = DifFim - DifIni
    if(Dif> MaiorDif):
        MaiorDif = Dif
        PriAnt = lstPrimos[j-1]
        PriPrx = lstPrimos[j]
    endif
    print(f'{DifFim} - {DifIni}: {Dif}')
    j = j + 1
endwhile
fim = time.time() #desliga o cronômetro
tempo = fim - inicio
tempo = int(tempo*100+0.50)/100
print('')
print(f'Maior intervalo entre dois primos consecutivos [{PriPrx}-{PriAnt}]: {MaiorDif}')
print('')
print(f'Tempo de processamento para o intervalo: {Lim1} a {Lim2}: {tempo} seg')
#FimPrograma-----

```

Entre com o limite inferior: 1  
Entre com o limite superior: 1000000000

2  
3  
5  
7  
11  
13  
17  
19  
23  
29  
31  
37  
41  
..

..... - ..... ..  
999999587 - 999999541: 46  
999999599 - 999999587: 12  
999999607 - 999999599: 8  
999999613 - 999999607: 6  
999999667 - 999999613: 54  
999999677 - 999999667: 10  
999999733 - 999999677: 56  
999999739 - 999999733: 6  
999999751 - 999999739: 12  
999999757 - 999999751: 6  
999999761 - 999999757: 4  
999999797 - 999999761: 36  
999999883 - 999999797: 86  
999999893 - 999999883: 10  
999999929 - 999999893: 36  
999999937 - 999999929: 8

Maior intervalo entre dois primos consecutivos [436273291-436273009: 282

Tempo de processamento para o intervalo: 1 a 1000000000: 159312.51 seg

Process finished with exit code 0