

Programar e Codificar - Diferença

Mário Leite

...

Descobri neste vídeo do You Tube (vide link <https://www.youtube.com/watch?v=UrxuCeiyFw&t=320s> - acesso em 07/11/2020 - 12:59), mais uma curiosidade sobre os números primos, e resolvi transformar o assunto em um programa, com o intuito de mostrar, mais uma vez, a diferença entre **Programar** e **Codificar**, e contribuir para o aprendizado, correto, da Programação. Nesse vídeo o professor apresenta a sequência dos primeiros onze números primos: **2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31**, e mostra que a soma dos quadros de **17, 19, 23, e 29** dá como resultado: **2020** (o ano atual). Entretanto, ele não fala que esta é uma coincidência e nem se existiria outras sequências de quatro primos, cuja soma de seus quadrados dê um ano tal como **2020**, de quatro dígitos, como: **3030, 4040, 5050**, e por aí vai...

Assim, resolvi investigar, criando o programa **"VerificaAno"**, cujo Algoritmo está mostrado na **figura 1**, com seu correspondente Pseudocódigo (posteriormente testado no Visualg) na **figura 2**. A **figura 3** mostra o código do programa, em Python, cuja saída pode ser analisada na **figura 4**. Na identificação do código do programa pode ser observado que destaquei: *Programação* (minha) e *Codificação* (de Clésio Matias), enfatizando que o programa foi feito a duas mãos: planejamento da solução (programação) e codificação, de maneira S-E-P-A-R-A-D-A, mostrando que *programar* é uma coisa e *codificar* é outra. Programei a solução do problema e passei o algoritmo para o Clésio, que sendo especialista em Python, o codificou nessa linguagem. Então, embora a codificação tenha ficado excelente, ela poderia ser feita em qualquer outra linguagem, mas, seguindo a orientação da ideia geral da solução dada pelo algoritmo (que rabisquei numa folha de papel - como mostra a **figura 1**) e formalizado em pseudocódigo (na **figura 2**).

Com esta postagem eu quis mostrar, mais uma vez, que o programador não deve se preocupar com a codificação sem, ANTES, saber COMO resolver o problema. E, embora o programa possa ter sua eficiência melhorada (principalmente no trecho "Verifica soma de quatro primos"), ou mesmo no trecho para "Gerar os primos na faixa", é assim que o programador deve se comportar. Isto quer dizer que, ao se sentar em frente ao computador para codificar o programa usando algum IDE, o programador deve ter em mãos o pseudocódigo da solução; senão, vai ficar olhando para o computador e o computador olhando para ele, sem rolar nada!

"Que a Lógica esteja com vocês; e não se deixem seduzir pelo lado sombrio da Codificação...!"

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

$$\hookrightarrow 17^2 + 19^2 + 23^2 + 29^2 = 2020$$

\Rightarrow Soma dos quadrados de quatro primos consecutivos.

Pesquisar outras faixas

- 1) Definir a faixa.
- 2) Gerar os primos da faixa.
- 3) Varrer a faixa: desde 2 até o último primo
- 4) ~~Mostrar~~ Calculando a soma dos quadrados a partir da posição $(i+3)$
- 4) ~~Mostrar~~ Se achar: N O N O ($N=2,3,4, \dots$)
Mostrar
Sinal
Voltar ao passo 3 com $(i+1)$
- 5) Mostrar o resultado da pesquisa:
Se (achou) ou Não (achou)
- 6) Mostrar os primos da faixa se o usuário desejar
- 7) FIM

Figura 1 - Algoritmo da solução do problema

Programa "VerificaAno"

```
//Mostra os anos do tipo 2020, 3030, 4040, 5050, ...  
//formados pela soma dos quadrados de quatro primos sequenciais numa faixa.  
//Programação: Mário Leite  
//-----
```

```
    Declare i, j, cont, contAno, lim, num, resto, soma, tamSoma: inteiro  
           dig1, dig2, dig3, dig4, resp, somaS: caractere  
           ehPrimo, cond, achou: logico  
           VetPrimos: array[1..1000] de inteiro
```

```
//-----
```

Início

```
    Escreva("Limite da faixa de pesquisa [1 a 1000000]: ")  
    Leia(lim)  
    EscrevaLn("") //salta uma linha  
    {Gera o vetor de primos: de 2 até o limite desejado}  
    cont <- 0 //inicializa contador de primos na faixa  
    Para num De 2 Até lim Faça  
        ehPrimo <- Verdadeiro  
        Para j De 2 Até (num -1) Faça  
            resto <- (num Mod j)  
            Se(resto=0) Então  
                ehPrimo <- Falso  
            FimSe  
        FimPara  
        Se(ehPrimo) Então  
            //num é primo  
            cont <- cont + 1 //acumula quantidade de primos  
            VetPrimos[cont] <- num  
        FimSe  
    FimPara  
    EscrevaLn("")  
    //-----
```

```

{Verifica soma de quatro primos}
achou <- Falso
Para i De 1 Até cont Faça
    soma <- 0
    Para j De i Até (i+3) Faça
        soma <- soma + Int(VetPrimos[j]^2) //soma quadrados dos primos
    FimPara
    tamSoma <- Tamanho(NumCarac(soma))
    Se(tamSoma=4) Então //tamanho de soma é compatível com Ano
        dig1 <- Copia(NumCarac(soma),1,1)
        dig2 <- Copia(NumCarac(soma),2,1)
        dig3 <- Copia(NumCarac(soma),3,1)
        dig4 <- Copia(NumCarac(soma),4,1)
        cond <- ((dig1=dig3) e (dig2=dig4))
        Se(cond) Então
            achou <- Verdadeiro
            EscrevaLn("Ano:", Soma)
            contAno <- contAno + 1
        FimSe
    FimSe
FimPara
//-----
Se(contAno=1) Então
    EscrevaLn("Apenas um ano encontrado na faixa: 2-",lim)
FimSe
Se(Nao(achou)) Então
    EscrevaLn("Nenhum Ano encontrado na faixa: 2-",lim)
FimSe
//-----
EscrevaLn("")
Escreva("Deseja ver os primos detectados na faixa 2-",lim,"[S/N]: ")
Leia(resp)
Se((resp="S") ou (resp="s")) Então
    EscrevaLn("Primos detectados na faixa 2-",lim,":")
    Para j De 1 Ate cont Faça
        EscrevaLn(VetPrimos[j])
    FimPara
    EscrevaLn("Quantidade de primos detectada: ", cont)
FimSe
FimPrograma
//-----

```

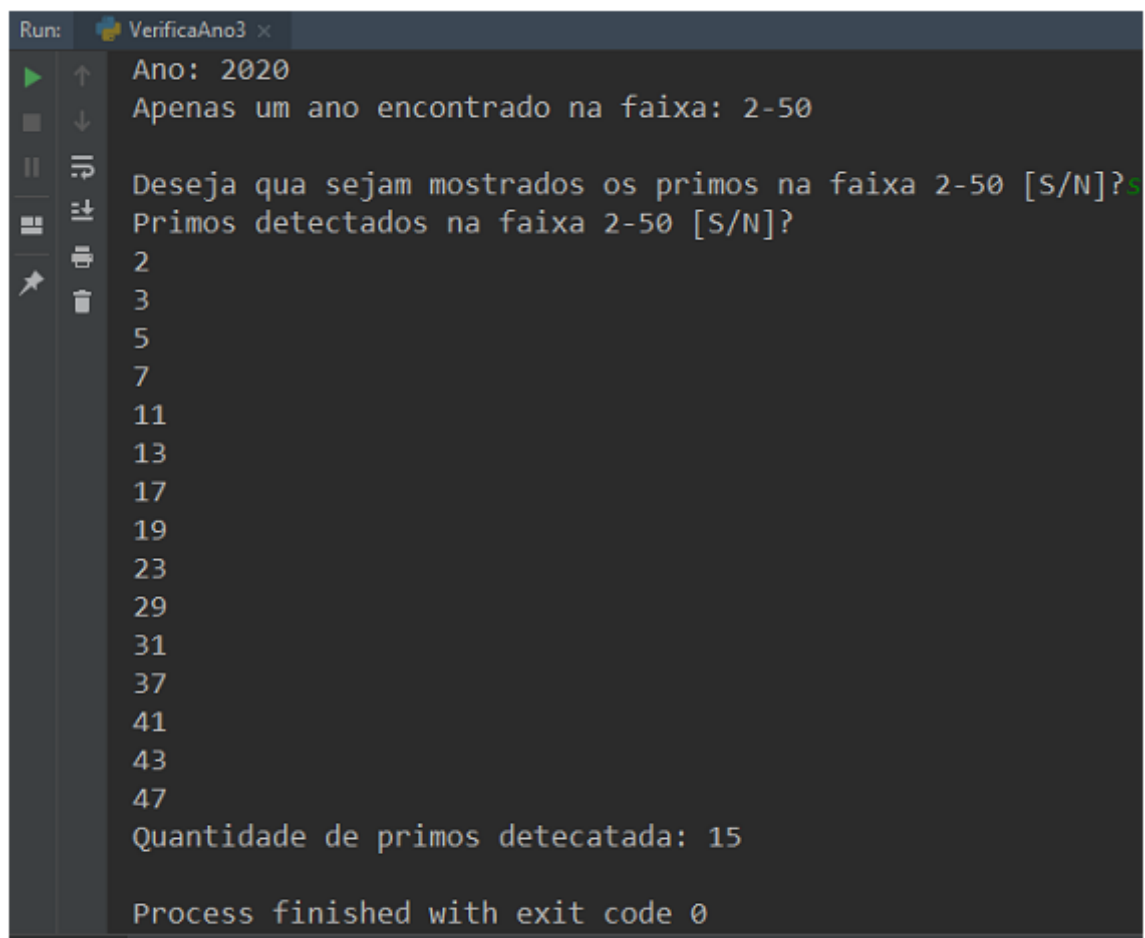
Figura 2 - Pseudocódigo do programa

```

'''
Programa "VerificaAno"
Mostra os anos do tipo 2020, 3030, 4040, 5050, ...
formados pela soma dos quadrados de quatro primos sequenciais, numa faixa.
Programação: Mário Leite
Codificação: Clésio Matias '''
#-----
'''ESCOPO GLOBAL DE VARIÁVEIS, CITADO AQUI APENAS PARA
MANTER A INTEGRIDADE E FIDELIDADE AO PSEUDOCÓDIGO'''
i, j, cont, contAno, lim, num, resto, soma, tamSoma= 0,0,0,0,0,0,0,0,0
dig1, dig2, dig3, dig4, resp, somaS = '', '', '', '', '', ''
ehPrimo, cond, achou = False, False, False,
VetPrimos=[] #inicializa o vetor (lista) de núemros primos
#-----
lim=int(input("Limite da faixa de pesquisa [1 a 1000000]: "))
print() #salta linha
#Gera o vetor de primos: de 2 até o limite desejado:
cont =0 #inicia o cobrador de primos na faixa
for num in range(2,lim+1):
    ehPrimo=True
    for j in range(2,num-1):
        resto = num%j
        if resto==0:
            ehPrimo=False
    if ehPrimo:
        cont+= 1 #acumula a quantidade de primos
        VetPrimos.append(num)
#-----
'''Verifica soma de quatro primos'''
achou =False
for i in range(1,cont):
    soma=0
    for j in range(i,i+4):
        #condicional acrescentada para evitar o estouro de pilha
        if j<len(VetPrimos):
            soma +=int(VetPrimos[j]**2) #soma o quadrado dos primos
    tamSoma = str(soma)
    if len(tamSoma)==4:#se a variavel soma tem 4 digitos
        dig1 = tamSoma[0]
        dig2 = tamSoma[1]
        dig3 = tamSoma[2]
        dig4 = tamSoma[3]
        cond = (dig1==dig3) and (dig2==dig4)
        if cond:
            achou=True
            print(f'Ano: {soma}')
            contAno+=1
#-----
#Analisa o resultado
if contAno==1:
    print(f'Apenas um ano encontrado na faixa: 2-{lim}')
if not achou:
    print(f'Nenhum ano encontrado na faixa: 2-{lim}')
#-----
#Decide se mostra, ou não, os primos detectados na faixa
print()
resp = input(f'Deseja que sejam mostrados os primos na faixa 2-{lim} [S/N]?')
if((resp=="S") or (resp=="s")):
    print(f'Primos detectados na faixa 2-{lim} [S/N]?')
    for j in range(0,cont):
        print(VetPrimos[j])
    print(f'Quantidade de primos detecatada: {cont}')
#-----Fim Programa -----

```

Figura 3 - Codificação do programa em Python



The image shows a terminal window titled "Run: VerificaAno3 x". The output of the program is as follows:

```
Ano: 2020
Apenas um ano encontrado na faixa: 2-50

Deseja que sejam mostrados os primos na faixa 2-50 [S/N]?
Primos detectados na faixa 2-50 [S/N]?
2
3
5
7
11
13
17
19
23
29
31
37
41
43
47
Quantidade de primos detectada: 15

Process finished with exit code 0
```

Figura 4 - Um exemplo de saída do programa