

Calculando PI com várias decimais

Mário Leite

...

A famosa constante **PI** - razão entre o perímetro de uma circunferência e seu diâmetro - que também é conhecida como “constante circular”, é uma das constantes universais mais importantes. Para QUALQUER objeto perfeitamente circular, dividindo seu perímetro pelo dobro de seu raio o valor encontrado sempre será, aproximadamente, o mesmo: **3.14**. Este é o valor muito utilizado para calcular áreas não retangulares; mas, neste caso a precisão não é boa; o ideal é usar PI com muitas decimais. Em cálculos de viagens espaciais a NASA utiliza o **PI** com quinze decimais (3.141592653589793), mas, para calcular áreas astronômicas ocupadas por conglomerados de galáxias, ou cálculo de paralaxe, por exemplo, pode ser necessário um número maior de decimais. Existem muitas fórmulas e métodos para calcular o valor de **PI**, cujo número de decimais conhecido, atualmente, é de impressionantes 8.000.000.000.000.000 (oito quatrilhões); resultado este obtido com *cluster* de computadores da Universidade Santa Clara na Califórnia (EUA), em 2013.

Apenas como exercício para testar a lógica do *loop Enquanto...FimEnquanto*, escrevi o programa “CalculaPI-BBP”, codificado em **Python**, para mostrar as potencialidades desta linguagem em cálculos com números flutuantes, baseando no pseudocódigo mostrado abaixo. O programa mostrar o valor de **PI** com um determinado número de decimais pela fórmula **BBP**. Esta fórmula foi apresentada em 1995 por David Harold **Bailey** em colaboração com Peter **Borwein** e Simon **Plouffe**; daí a sigla com as letras iniciais dos sobrenomes de seus criadores. Embora a codificação mostrada aqui tenha sido feita em Python, os programadores de outras linguagens podem criar seus códigos baseando no pseudocódigo, fazendo as adaptações necessárias.

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

Programa "CalculaPI-BBP"

```
//Calcula PI com determinado número de decimais através da "Fórmula BBP"  
//Em Pseudocódigo  
//Autor: Mário Leite  
//-----
```

```
Declare d, x, k: inteiro  
Soma, fator, Parc1, Parc2, Parc3, Parc4: real  
SomaS: caractere  
Verdade: lógico
```

Início

```
d ← 0  
Enquanto ((d<2) ou (d>1000)) Faça  
    Escreva("Digite o número de decimais para PI [2 a 1000]: ")  
    Leia(d)  
FimEnquanto  
EscrevaLn("") //salta uma linha
```

```
Soma ← 0  
SomaSAnt ← "  
Verdade ← .V.  
k ← 0
```

```
Enquanto (Verdade) Faça //loop para calcular o valor de PI  
    fator ← Decimal((1/16)**k)  
    Parc1 ← Decimal(4/((8*k)+1))  
    Parc2 ← Decimal(2/((8*k)+4))  
    Parc3 ← Decimal(1/((8*k)+5))  
    Parc4 ← Decimal(1/((8*k)+6))  
    Soma ← Decimal(Soma + fator*(Parc1 - Parc2 - Parc3 - Parc4))
```

```

SomaS ← NumCarac(Soma) //converte Soma para string
Se(SomaS = SomaSAnt) Então
    Verdade ← .F.
    Abandone //abandona o loop: atingiu o valor de PI desejado
FimSe
EscrevaLn ("PI =", SomaS)
k = k + 1
SomaSAnt ← SomaS //nova SomaS para a próxima iteração
FimEnquanto
EscrevaLn ("" )
EscrevaLn("PI com " , d, " decimais: " , SomaS)
EscrevaLn("Valor alcançado com " , k, " iterações.")
FimPrograma

```

```

'''
Programa "CalculaPI-BBP"
Calcula PI com um determinado número de decimais através da "Fórmula BBP"
Autor: Mário Leite
-----
'''

from decimal import Decimal, getcontext, setcontext #biblioteca para aumentar decimais
ctx = getcontext()
endwhile = "endwhile"
endif = "endif"

d = 0
while((d<2) or (d>1000)):
    d= int(input("Digite o número de decimais desejado [2 a 1000]: "))
endwhile
print()

ctx.prec = d+1 #define a precisão em função do número de decimais
setcontext(ctx)

Soma = 0
SomaSAnt = ""
Verdade = True
k = 0
while (Verdade): #Loop para calcular o valor de PI
    fator = Decimal((1/16)**k)
    Parc1 = Decimal(4/((8*k)+1))
    Parc2 = Decimal(2/((8*k)+4))
    Parc3 = Decimal(1/((8*k)+5))
    Parc4 = Decimal(1/((8*k)+6))
    Soma = Decimal(Soma + fator * (Parc1 - Parc2 - Parc3 - Parc4))
    SomaS = str(Soma) #converte Soma em string
    if(SomaS==SomaSAnt):
        Verdade = False
        break #abandona o loop: atingiu o valor de PI desejado
    endif
    print(f'PI = {SomaS}')
    k = k + 1
    SomaSAnt = SomaS #redefine SomaSAnt para a próxima iteração
endwhile
print()
print(f'PI com {d} decimais: {SomaS}')
print(f'Valor alcançado com {k} iterações')
#FimPrograma-----

```

"D:\Cantinho da Programação\Códigos\Python\PIBBP2\venv\Scripts\python.exe" C:/Users/Usuario/.PyCharmCE2018.3
Digite o número de decimais desejado [2 a 1000]: 100

```
PI = 3.133333333333333314829616256247390992939472198486328125
PI = 3.14142246642246641963325259894190821796655654907226562500000
PI = 3.1415873903465815201898996000817732010546023957431316375732421875
PI = 3.141592457567435378976803826873986924539394749444909393787384033203125
PI = 3.14159264546033631669691828453352955285993175493786111474037170410156250
PI = 3.141592653228087531874276889108624937797797915806086166412569582462310791015625
PI = 3.1415926535728808249251392041336991163276264049653008214590954594314098358154296875
PI = 3.14159265358897270208067619510884110084033746766729588983935173018835484981536865234375
PI = 3.1415926535897522723760762972900385274492984636392906017121617878729011863470077514648437500
PI = 3.1415926535897911435286753947732936302056600738608332229473418806264817249029874801635742187500
PI = 3.1415926535897931267540689929032952142660825870284496273872332228194181880098767578601837158203125
PI = 3.1415926535897932298511906907919568209091883327625569936324980389874550468221059418283402919769287109
PI = 3.1415926535897932352946647513637392667920470326335825921318415331322942973812928357801865786314010620
PI = 3.1415926535897932355858759308021580813850592011995254829986059522865891185938735929994436446577310562
PI = 3.1415926535897932356016309108998588870651495190893702977235645216520735904276728778228289229446090758
PI = 3.1415926535897932356024916035325592844795916455287331072415464452552785284923306252657937420735834167
PI = 3.1415926535897932356025390240000048467323648877433278372963392330696531720975891673920676883113856093
PI = 3.1415926535897932356025416562866988599065198494532111288023452721140414894593433037196707896532288374
PI = 3.1415926535897932356025418033776378876393634091891116036788805430406934456593734083735874797888021349
PI = 3.1415926535897932356025418116459679159157629261745198592587131638232099025695587682531185485146525694
PI = 3.1415926535897932356025418121132390175510466360378116596486947446460065102430440528051597524871461128
PI = 3.1415926535897932356025418121397738765655458783222586171992247294854437179489820423038528803491535856
PI = 3.1415926535897932356025418121412873313616211916689034120136889950883006677845122984463585622118948199
PI = 3.1415926535897932356025418121413739996472246697492896460915070037888132401146022233693715368187329083
PI = 3.1415926535897932356025418121413789809540400008353938890522876269577708633529631337629793550265638390
PI = 3.1415926535897932356025418121413792682242319749935830548837136610016010607033817249448068195779883870
PI = 3.1415926535897932356025418121413792848426144198861105705132948548129714487536188391586980022254570885
PI = 3.1415926535897932356025418121413792858067480471731863489938651127192556164932816513906224828641058070
PI = 3.1415926535897932356025418121413792858628329835933646566136258376507406942924416183089526933611453587
PI = 3.1415926535897932356025418121413792858661036429409533487878538756817775542899859420320221515477535966
PI = 3.1415926535897932356025418121413792858662948183896398528340520775099695792871579954117005490187658465
PI = 3.1415926535897932356025418121413792858663060171855002861129883550312160174647561367389785605030942514
PI = 3.1415926535897932356025418121413792858663066745311238511006748539146472466611847775312063065388530134
PI = 3.1415926535897932356025418121413792858663067131896860620654582111292253839649638359531221053732689749
PI = 3.1415926535897932356025418121413792858663067154672894206606044460155187233577671906439145023651963012
```



```

PI = 3.14159265358979323560254181214137928586630671561017039405107356104172584064382390904417948901529799518
PI = 3.1415926535897932356025418121413792858663067156096492155346113526560859738784006196030172755819219690
PI = 3.1415926535897932356025418121413792858663067156101195736591224184303209378253230912989713638364070134
PI = 3.1415926535897932356025418121413792858663067156101474586575291909495332702027780187610430357746699799
PI = 3.1415926535897932356025418121413792858663067156101491140570683995898733197945767330315405944863534907
PI = 3.1415926535897932356025418121413792858663067156101492124571937810355704251615723231572585869601286936
PI = 3.1415926535897932356025418121413792858663067156101492183134712884978522146510939033451192528971625223
PI = 3.1415926535897932356025418121413792858663067156101492186624149746364227088332118780122261215459098739
PI = 3.1415926535897932356025418121413792858663067156101492186832298133311175518370348027674874824252376950
PI = 3.1415926535897932356025418121413792858663067156101492186844727605672043789586668437333289254223874088
PI = 3.1415926535897932356025418121413792858663067156101492186845470579781102147563611518278604691940068022
PI = 3.1415926535897932356025418121413792858663067156101492186845515034386865626390659508177556237183063824
PI = 3.1415926535897932356025418121413792858663067156101492186845517696729383103464614343976111926875479200
PI = 3.1415926535897932356025418121413792858663067156101492186845517856316668875445051648457432539060977301
PI = 3.1415926535897932356025418121413792858663067156101492186845517865890901953740447387974941903644170827
PI = 3.1415926535897932356025418121413792858663067156101492186845517866465767333917840250241865903758914202
PI = 3.1415926535897932356025418121413792858663067156101492186845517866500311184749408687219653045431551529
PI = 3.1415926535897932356025418121413792858663067156101492186845517866502388510351315789534999276758829729
PI = 3.1415926535897932356025418121413792858663067156101492186845517866502513523349461865275051653069488900
PI = 3.1415926535897932356025418121413792858663067156101492186845517866502521051889753109445339565271403236
PI = 3.1415926535897932356025418121413792858663067156101492186845517866502521505580872998628557630104896706
PI = 3.1415926535897932356025418121413792858663067156101492186845517866502521532939427560443943613575932111
PI = 3.1415926535897932356025418121413792858663067156101492186845517866502521534590246417224519383562880759
PI = 3.1415926535897932356025418121413792858663067156101492186845517866502521534689917607903861108412197315
PI = 3.1415926535897932356025418121413792858663067156101492186845517866502521534695938971550025449713941409
PI = 3.1415926535897932356025418121413792858663067156101492186845517866502521534696302942340164730903980773
PI = 3.1415926535897932356025418121413792858663067156101492186845517866502521534696324955205320087211682346
PI = 3.1415926535897932356025418121413792858663067156101492186845517866502521534696326287245346750230388025
PI = 3.1415926535897932356025418121413792858663067156101492186845517866502521534696326367891062168500643382
PI = 3.1415926535897932356025418121413792858663067156101492186845517866502521534696326372776029354005224156
PI = 3.1415926535897932356025418121413792858663067156101492186845517866502521534696326373072070216119364041
PI = 3.1415926535897932356025418121413792858663067156101492186845517866502521534696326373090019410516997610
PI = 3.1415926535897932356025418121413792858663067156101492186845517866502521534696326373091108178829575575
PI = 3.1415926535897932356025418121413792858663067156101492186845517866502521534696326373091174250818893211
PI = 3.1415926535897932356025418121413792858663067156101492186845517866502521534696326373091178262118795835
PI = 3.1415926535897932356025418121413792858663067156101492186845517866502521534696326373091178505750275555
PI = 3.1415926535897932356025418121413792858663067156101492186845517866502521534696326373091178520553526384
PI = 3.1415926535897932356025418121413792858663067156101492186845517866502521534696326373091178521453337463
PI = 3.1415926535897932356025418121413792858663067156101492186845517866502521534696326373091178521508053097
PI = 3.1415926535897932356025418121413792858663067156101492186845517866502521534696326373091178521511381477
PI = 3.1415926535897932356025418121413792858663067156101492186845517866502521534696326373091178521511584017
PI = 3.1415926535897932356025418121413792858663067156101492186845517866502521534696326373091178521511596346
PI = 3.1415926535897932356025418121413792858663067156101492186845517866502521534696326373091178521511597097
PI = 3.1415926535897932356025418121413792858663067156101492186845517866502521534696326373091178521511597143
PI = 3.1415926535897932356025418121413792858663067156101492186845517866502521534696326373091178521511597146
PI com 100 decimais: 3.1415926535897932356025418121413792858663067156101492186845517866502521534696326373091178521511597146
Valor alcançado com 80 iterações
Process finished with exit code 0

```

Saída do programa "CalculaPI-BBP": PI com 100 decimais após 80 iterações da fórmula BBP