

Computación

Aula Virtual: <https://famaf.aulavirtual.unc.edu.ar/course/view.php?id=747>

Resguardo tutoriales: <https://www.famaf.unc.edu.ar/~moreschi/docencia/Computacion/>

Tutorial Problema 7 de la Guía N° 2

Problema 7: Haga un programa que, para un dado $x > 1$, permita calcular una aproximación al valor de $\log(x)$ con un error menor o igual a 10^{-3} . A tal fin recuerde la serie de Taylor

$$\log(1 + s) = - \sum_{n=1}^{\infty} (-1)^n \frac{s^n}{n}, \quad (\text{logaritmo natural})$$

y el teorema del error para series alternantes truncadas. El programa debe retornar: La cantidad de términos sumados, el valor aproximado de $\log(x)$, una cota para el error de la aproximación obtenida y el valor de $\log(x)$ dado por el módulo `math`. Haga el cálculo para $x = 1.1$, $x = 2.0$ y $x = 11.0$.

Tutorial:

- Guarde en el archivo `p7-g2.py` las siguientes instrucciones:

```

1 import math
2
3 #-----
4 x = 1.1
5 s = x - 1.
6 logx = math.log(x)
7
8 suma=0
9 error = 1.
10 n=1
11
12 while error > 10**(-3):
13     suma = suma - (-1)**n * s**n/n
14     n = n +1
15     error = abs(suma - logx)
16     if (n==1000):
17         print("demasiadas iteraciones")
18         quit()
19     #print(" s= ",s)
20
21 print("          x = ",x )
22 print("          error = %35.22e " % (error) )
23 print(" aprx log(x) = %35.22e " % (suma) )
24 print(" math.log(x) = %35.22e " % (math.log(x)) )
25 print("          n = ",n )
26
27 #-----
28 x = 2.0
29 s = x - 1.

```

```

30 logx = math.log(x)
31
32 suma=0
33 error = 1.
34 n=1
35
36 while error > 10**(-3):
37     suma = suma - (-1)**n * s**n/n
38     n = n +1
39     error = abs(suma - logx)
40     if (n==1000):
41         print("demasiadas iteraciones")
42         quit()
43     #print("  s= ",s)
44
45 print()
46 print("          x = ",x )
47 print("      error = %35.22e " % (error) )
48 print("  aprx log(x) = %35.22e " % (suma) )
49 print("  math.log(x) = %35.22e " % (math.log(x)) )
50 print("          n = ",n )
51
52 #-----
53 x = 11.
54 s = x - 1.
55 logx = math.log(x)
56 print()
57 print("          x = ",x )
58
59 suma=0
60 error = 1.
61 n=1
62
63 # Lo que sigue le dará error al correrlo; pues los
64 # términos divergen.
65 while error > 10**(-3):
66     suma = suma - (-1)**n * s**n/n
67     n = n +1
68     error = abs(suma - logx)
69     #print("n = ",n,"suma = ",suma)
70     if (n==1000):
71         print("demasiadas iteraciones")
72         quit()
73
74 print("      error = %35.22e " % (error) )
75 print("  aprx log(x) = %35.22e " % (suma) )
76 print("  math.log(x) = %35.22e " % (math.log(x)) )
77 print("          n = ",n )

```

- Desde la terminal ejecute:
python3 p7-g2.py
o suba de a poco los comandos en una ventana
ipython3
e interprete el resultado.