Tarea Método de la Secante

Angel Caceres Licona

May 28, 2020

1 Termine las iteraciones y verifique sus resultados con la tabla 2

Yo lo que hice fue programar el método y correrlo con los mismos datos. Obtuve resultados idénticos:

```
Iteracion          a                    b                    c                    f(c)                     error
0          0.000000000000       1.000000000000       0.678614100575       0.120395183590          1.000000000000
1          1.000000000000       0.678614100575       0.569062251401       -0.027213687998          0.321385899425
2          0.678614100575       0.569062251401       0.589259613598       0.001007800210          0.109551849174
3          0.569062251401       0.589259613598       0.588538358018       0.000007786074          0.020197362197
4          0.589259613598       0.588538358018       0.588532742348       -0.000000002266          0.000721255581
5          0.588538358018       0.588532742348       0.588532743982       0.000000000000          0.000005615670
La raiz buscada es: 0.588532743982 con 6 iteraciones.
```

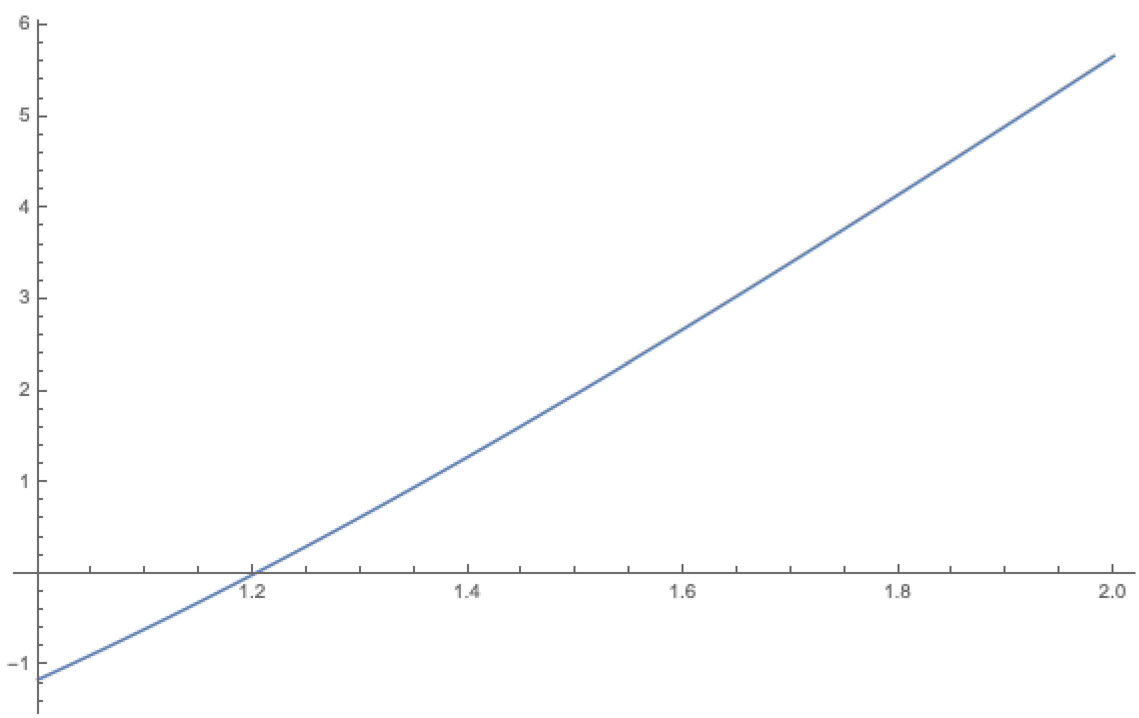
2 Código del programa

```
from math import *

def fx(x):
    respuesta = -8*exp(1-x) + 7/x
    return respuesta


def secante(a,b,tol):
    fa = fx(a)
    fb = fx(b)
    c = b - fb*((b-a)/(fb-fa))
    i = 1
    error = abs(b-a)
    fc = fx(c)
    print "Iteracion          a                    b                  c                    f(c)                    error "
    for i in range (1000):
        c = b - (fb*(b-a)/(fb-fa))
        fc = fx(c)
        print "%.0f" %i, "           %.12f" %a,"           %.12f" %b,"    %.12f" %c, "    %.12f" %fc, "         %.12f" %error
        i = i+1
        if (fc==0.0 or abs(b-a) < tol):
```

```
                break

            a = b
            b = c
            fa = fx(a)
            fb = fx(b)

            error = abs(b-a)

        print "La raiz buscada es: %.12f" %c, "con " + str(i) + " iteraciones."

    secante(0.5,0.6,0.000000005)
```

3 Considerar la función $f(x) = x^2 - 4\cos(x), x \in \mathbb{R}$...

3.1 Graficar $f(x)$ en el intervalo $1, 2)$.



3.2 Usar el método de la secante para localizar una aproximacion...

Usando mi programa obtuve los siguientes resultados:

n	p_{n-1}	p_n	p_{n+1}	$\lvert p_n - p_{n-1} \rvert$	$f(x_n)$
0	1	2	1.170120690182	1	-0.190979794047
1	2	1.170120690182	1.170120690182	1.197187270733	0.829879309818
2	1.170120690182	1.197187270733	1.201577567782	-0.190979794047	0.027066580551
3	1.197187270733	1.201577567782	1.201538251243	-0.026654203791	0.004390297049
4	1.201577567782	1.201538251243	1.201538299340	0.000240853997	0.000039316539

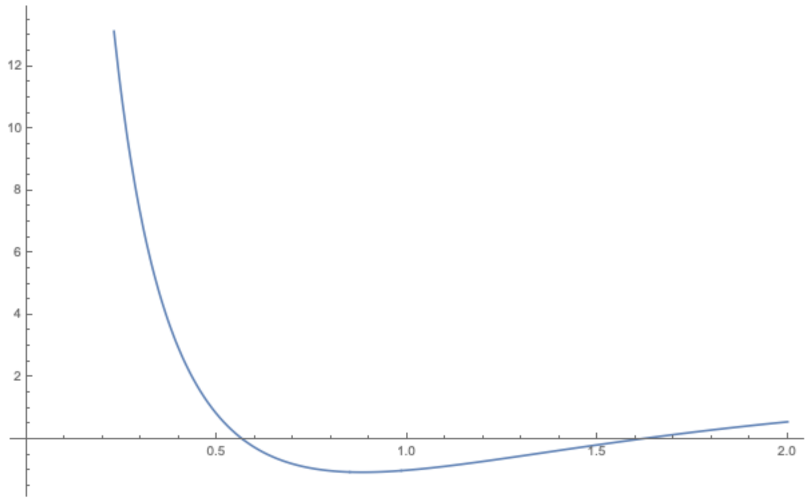
3.3 Aplicar el método de bisección con la misma tolerancia...

Se requirieron 29 iteraciones para obtener un resultado similar, por lo que vemos que el método de bisección es mucho mas lento.

```
Iteracion     a     b     c     f(c)
('1.0000000', '1.0000000', '    2.0000000', '    1.5000000', '    4.7795512')
('2.0000000', '1.0000000', '    1.5000000', '    1.2500000', '    1.1801168')
('3.0000000', '1.0000000', '    1.2500000', '    1.1250000', '    -0.1228994')
('4.0000000', '1.1250000', '    1.2500000', '    1.1875000', '    0.4926221')
('5.0000000', '1.1250000', '    1.1875000', '    1.1562500', '    0.1762403')
('6.0000000', '1.1250000', '    1.1562500', '    1.1406250', '    0.0245610')
('7.0000000', '1.1250000', '    1.1406250', '    1.1328125', '    -0.0496910')
('8.0000000', '1.1328125', '    1.1406250', '    1.1367188', '    -0.0126961')
('9.0000000', '1.1367188', '    1.1406250', '    1.1386719', '    0.0058995')
('10.0000000', '1.1367188', '    1.1386719', '    1.1376953', '    -0.0034065')
('11.0000000', '1.1376953', '    1.1386719', '    1.1381836', '    0.0012445')
('12.0000000', '1.1376953', '    1.1381836', '    1.1379395', '    -0.0010815')
('13.0000000', '1.1379395', '    1.1381836', '    1.1380615', '    0.0000813')
('14.0000000', '1.1379395', '    1.1380615', '    1.1380005', '    -0.0005001')
('15.0000000', '1.1380005', '    1.1380615', '    1.1380310', '    -0.0002094')
('16.0000000', '1.1380310', '    1.1380615', '    1.1380463', '    -0.0000640')
('17.0000000', '1.1380463', '    1.1380615', '    1.1380539', '    0.0000087')
('18.0000000', '1.1380463', '    1.1380539', '    1.1380501', '    -0.0000277')
('19.0000000', '1.1380501', '    1.1380539', '    1.1380520', '    -0.0000095')
('20.0000000', '1.1380520', '    1.1380539', '    1.1380529', '    -0.0000004')
('21.0000000', '1.1380529', '    1.1380539', '    1.1380534', '    0.0000041')
('22.0000000', '1.1380529', '    1.1380534', '    1.1380532', '    0.0000018')
('23.0000000', '1.1380529', '    1.1380532', '    1.1380531', '    0.0000007')
('24.0000000', '1.1380529', '    1.1380531', '    1.1380530', '    0.0000001')
('25.0000000', '1.1380529', '    1.1380530', '    1.1380530', '    -0.0000001')
('26.0000000', '1.1380530', '    1.1380530', '    1.1380530', '    -0.0000000')
('27.0000000', '1.1380530', '    1.1380530', '    1.1380530', '    0.0000001')
('28.0000000', '1.1380530', '    1.1380530', '    1.1380530', '    0.0000000')
('29.0000000', '1.1380530', '    1.1380530', '    1.1380530', '    0.0000000')
('La raiz buscada es: 1.1380530', 'con 29 iteraciones.')
```

4 Considere la función $f(x) = -8e^{1-x} + \frac{7}{x}$

4.1 Grafique en $(0, 2)$ y verifique que tiene dos raíces.



4.2 Poner los resultados de cada raíz en una tabla

Para la primer raíz tenemos:

n	p_{n-1}	p_n	p_{n+1}	$\lvert p_n - p_{n-1} \rvert$	$f(x_n)$
0	0.5	0.6	0.575149260929	-0.064143321099	1
1	0.6	0.575149260929	0.567327347390	0.007582973043	-0.064143321099
2	0.575149260929	0.567327347390	0.568154287638	-0.000182851406	0.007821913540
3	0.567327347390	0.568154287638	0.568134816790	-0.000000504095	0.000826940249
4	0.568154287638	0.568134816790	0.568134762964	0.000000000034	0.000019470848
5	0.568134816790	0.568134762964	0.568134762967	0.000000000000	0.000000053827

Para la segunda raíz tenemos:

n	p_{n-1}	p_n	p_{n+1}	$\lvert p_n - p_{n-1} \rvert$	$f(x_n)$
0	1.55	1.62	1.609561727949	0.000297514517	0.07
1	1.620000000000	1.609561727949	1.609380484711	-0.000000960168	0.010438272051
2	1.609561727949	1.609380484711	1.609381067755	0.000000000052	0.000181243238
3	1.609380484711	1.609381067755	1.609381067723	0.000000000000	0.000000583044

4.3 Aplicar el método de bisección con la misma tolerancia...

Se requirieron 26 iteraciones para obtener un resultado similar, para la 1er raiz:

```
Ingrese el valor a:0.5
Ingrese el valor b:0.6
Ingrese la tolerancia:0.000000005
Iteracion     a     b     c     f(c)
('1.0000000', '0.5000000', '    0.6000000', '    0.5500000', '    0.1807752')
('2.0000000', '0.5500000', '    0.6000000', '    0.5750000', '    -0.0628103')
('3.0000000', '0.5500000', '    0.5750000', '    0.5625000', '    0.0538021')
('4.0000000', '0.5625000', '    0.5750000', '    0.5687500', '    -0.0057501')
('5.0000000', '0.5625000', '    0.5687500', '    0.5656250', '    0.0237085')
('6.0000000', '0.5656250', '    0.5687500', '    0.5671875', '    0.0089006')
('7.0000000', '0.5671875', '    0.5687500', '    0.5679687', '    0.0015557')
('8.0000000', '0.5679687', '    0.5687500', '    0.5683594', '    -0.0021021')
('9.0000000', '0.5679687', '    0.5683594', '    0.5681641', '    -0.0002744')
('10.0000000', '0.5679687', '    0.5681641', '    0.5680664', '    0.0006404')
('11.0000000', '0.5680664', '    0.5681641', '    0.5681152', '    0.0001829')
('12.0000000', '0.5681152', '    0.5681641', '    0.5681396', '    -0.0000458')
('13.0000000', '0.5681152', '    0.5681396', '    0.5681274', '    0.0000686')
('14.0000000', '0.5681274', '    0.5681396', '    0.5681335', '    0.0000114')
('15.0000000', '0.5681335', '    0.5681396', '    0.5681366', '    -0.0000172')
('16.0000000', '0.5681335', '    0.5681366', '    0.5681351', '    -0.0000029')
('17.0000000', '0.5681335', '    0.5681351', '    0.5681343', '    0.0000043')
('18.0000000', '0.5681343', '    0.5681351', '    0.5681347', '    0.0000007')
('19.0000000', '0.5681347', '    0.5681351', '    0.5681349', '    -0.0000011')
('20.0000000', '0.5681347', '    0.5681349', '    0.5681348', '    -0.0000002')
('21.0000000', '0.5681347', '    0.5681348', '    0.5681347', '    0.0000002')
('22.0000000', '0.5681347', '    0.5681348', '    0.5681348', '    0.0000000')
('23.0000000', '0.5681348', '    0.5681348', '    0.5681348', '    -0.0000001')
('24.0000000', '0.5681348', '    0.5681348', '    0.5681348', '    -0.0000000')
('25.0000000', '0.5681348', '    0.5681348', '    0.5681348', '    -0.0000000')
('26.0000000', '0.5681348', '    0.5681348', '    0.5681348', '    0.0000000')
('La raiz buscada es: 0.5681348', 'con 26 iteraciones.')
```

Se requirieron 25 iteraciones para obtener un resultado similar, para la 2a raiz:

```
Ingrese el valor a:1.55
Ingrese el valor b:1.62
Ingrese la tolerancia:0.000000005
Iteracion     a     b     c     f(c)
('1.0000000', '1.5500000', '    1.6200000', '    1.5850000', '    -0.0404431')
('2.0000000', '1.5850000', '    1.6200000', '    1.6025000', '    -0.0113558')
('3.0000000', '1.6025000', '    1.6200000', '    1.6112500', '    0.0030762')
('4.0000000', '1.6025000', '    1.6112500', '    1.6068750', '    -0.0041304')
('5.0000000', '1.6068750', '    1.6112500', '    1.6090625', '    -0.0005247')
('6.0000000', '1.6090625', '    1.6112500', '    1.6101563', '    0.0012764')
('7.0000000', '1.6090625', '    1.6101563', '    1.6096094', '    0.0003760')
('8.0000000', '1.6090625', '    1.6096094', '    1.6093359', '    -0.0000743')
('9.0000000', '1.6093359', '    1.6096094', '    1.6094727', '    0.0001508')
('10.0000000', '1.6093359', '    1.6094727', '    1.6094043', '    0.0000383')
('11.0000000', '1.6093359', '    1.6094043', '    1.6093701', '    -0.0000180')
('12.0000000', '1.6093701', '    1.6094043', '    1.6093872', '    0.0000101')
('13.0000000', '1.6093701', '    1.6093872', '    1.6093787', '    -0.0000040')
('14.0000000', '1.6093787', '    1.6093872', '    1.6093829', '    0.0000031')
('15.0000000', '1.6093787', '    1.6093829', '    1.6093808', '    -0.0000004')
('16.0000000', '1.6093808', '    1.6093829', '    1.6093819', '    0.0000013')
('17.0000000', '1.6093808', '    1.6093819', '    1.6093813', '    0.0000004')
('18.0000000', '1.6093808', '    1.6093813', '    1.6093811', '    -0.0000000')
('19.0000000', '1.6093811', '    1.6093813', '    1.6093812', '    0.0000002')
('20.0000000', '1.6093811', '    1.6093812', '    1.6093811', '    0.0000001')
('21.0000000', '1.6093811', '    1.6093811', '    1.6093811', '    0.0000001')
('22.0000000', '1.6093811', '    1.6093811', '    1.6093811', '    0.0000000')
('23.0000000', '1.6093811', '    1.6093811', '    1.6093811', '    0.0000000')
('24.0000000', '1.6093811', '    1.6093811', '    1.6093811', '    0.0000000')
('25.0000000', '1.6093811', '    1.6093811', '    1.6093811', '    -0.0000000')
('La raiz buscada es: 1.6093811', 'con 25 iteraciones.')
```