UNIVERSIDAD DEL CAUCA

Presentado por:

NICOLLE ALEXANDRA MONTAÑO CIFUENTES

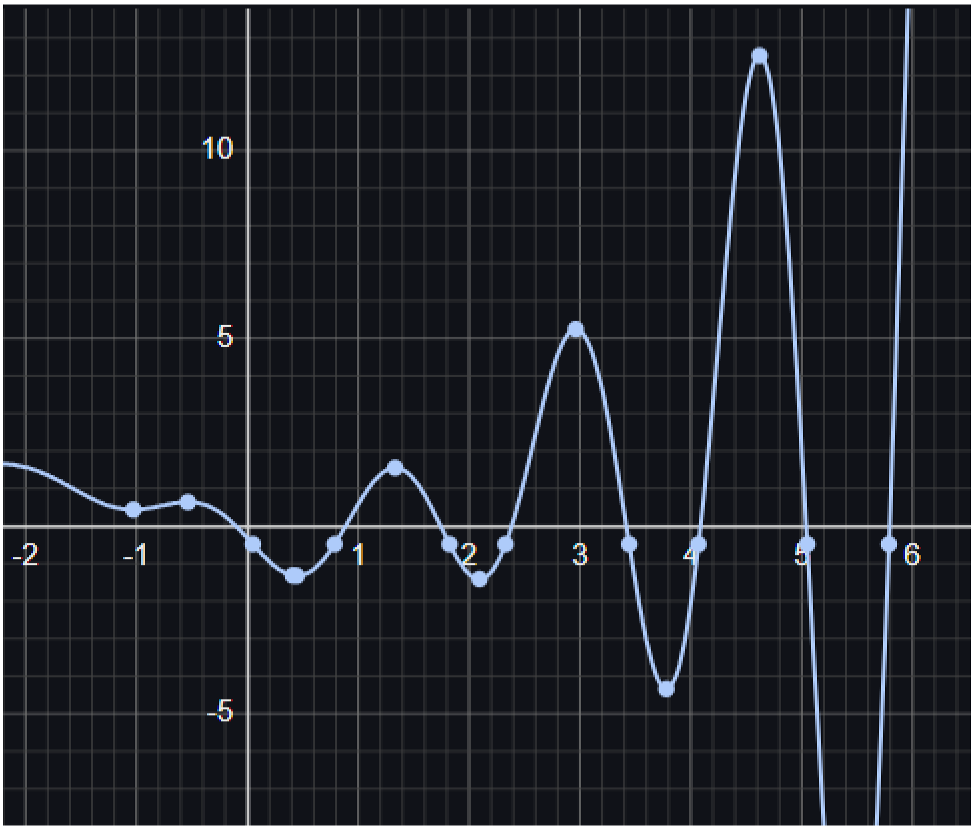
Presentado a:
CARLOS ALBERTO ARDILA

Materia:
ANÁLISIS NUMÉRICO

Trabajo:
PRIMER PARCIAL

$$f(x)=ln(x^2+1)-e^{(x/2)}cos(x\pi)$$

en el intervalo {-2,6}



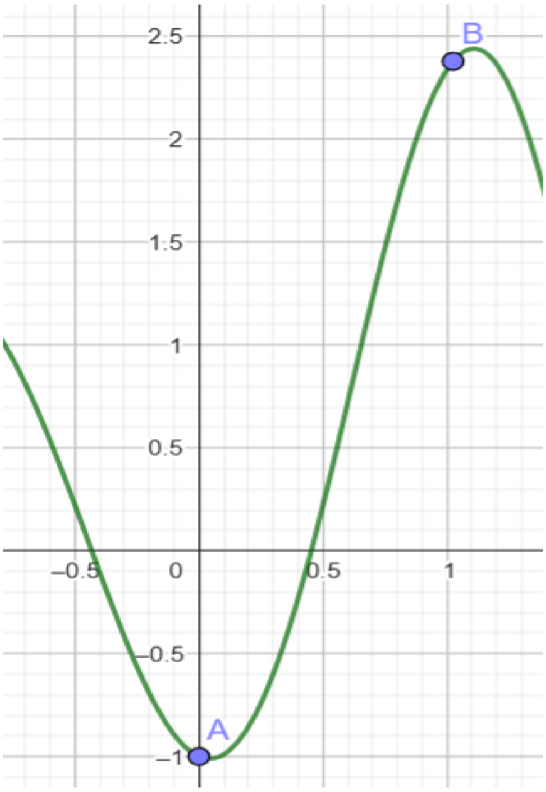
MÉTODO 1

BISECCIÓN

Procederemos a encontrar la raíz utilizando el método de Bisección. Para aplicar este método, es necesario cumplir con las siguientes condiciones:

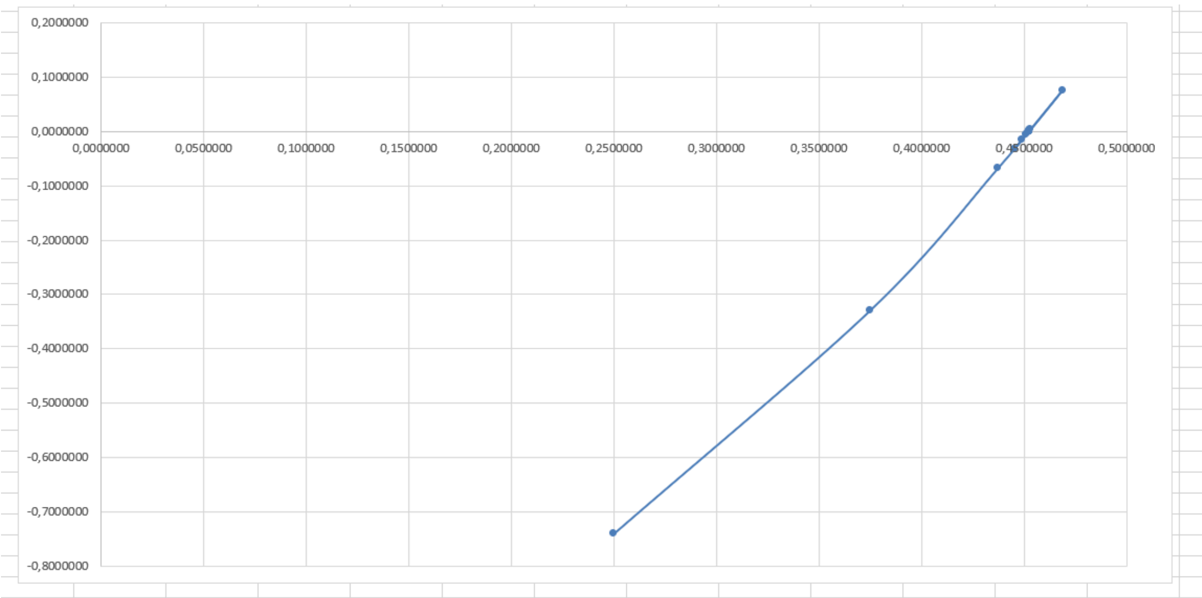
1. La función f(x)f(x)f(x) debe ser continua en el intervalo considerado.
2. Deben existir valores xa y xb tales que f(xa) · f(xb)<0 lo que garantiza un cambio de signo y, por lo tanto, la existencia de al menos una raíz en el intervalo.

Tomando en cuenta estos criterios y la gráfica , se seleccionó el intervalo [0,1] para encontrar la primera raíz, asegurando que cumple con las condiciones necesarias para la aplicación del método.



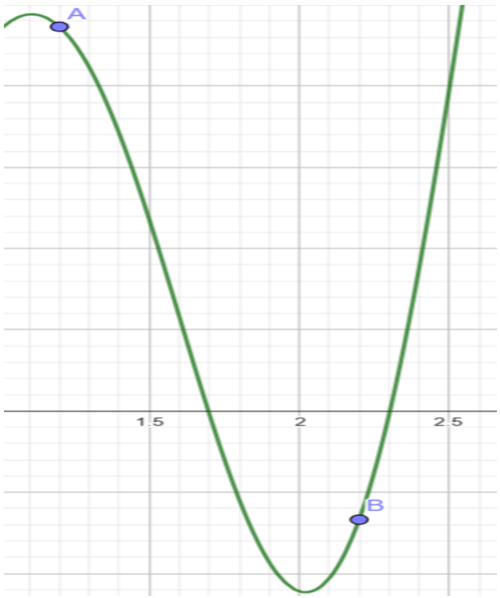
```
 - B I S E C C I O N -
Digite el extremo inferior (a) : 0
Digite el extremo superior (b) : 1
Digite el numero de iteraciones : 200

n       Raiz                    f(Raiz)                 Er
2       0.2500000000            -0.7406323334           1
3       0.3750000000            -0.3300279743           0.3333333333
4       0.4375000000            -0.06765949645          0.1428571429
5       0.4687500000            0.07472142924           0.06666666667
6       0.4531250000            0.002705725317          0.03448275862
7       0.4453125000            -0.03269167382          0.01754385965
8       0.4492187500            -0.01504562462          0.008695652174
9       0.4511718750            -0.006182980257         0.004329004329
10      0.4521484375            -0.00174186855          0.002159827214
11      0.4526367188            0.000481120189          0.001078748652
12      0.4523925781            -0.0006305764889                0.0005396654074
13      0.4525146484            -7.477869453e-05                0.0002697599137
14      0.4525756836            0.0002031581152         0.0001348617667
15      0.4525451660            6.418655178e-05         6.743543058e-05
16      0.4525299072            -5.296861069e-06                3.371885221e-05
17      0.4525375366            2.944464794e-05         1.685914187e-05
18      0.4525337219            1.207384408e-05         8.429641993e-06
19      0.4525318146            3.388479167e-06         4.214838761e-06
20      0.4525308609            -9.541940351e-07                2.107423822e-06
21      0.4525313377            1.217141795e-06         1.053710801e-06
22      0.4525310993            1.314736872e-07         5.268556779e-07
23      0.4525309801            -4.113602222e-07                2.634279083e-07
24      0.4525310397            -1.399432797e-07                1.317139368e-07
25      0.4525310695            -4.2347994e-09          6.585696407e-08
```

En este caso, la raíz se encuentra tras un total de **24 iteraciones.** Se puede observar que, aunque el método de Bisección garantiza la convergencia, no es el más eficiente en términos de iteraciones.Si bien la cota de error obtenida es buena, no es la mejor en comparación con otros métodos más rápidos.



(gráfica con los datos de desempeño del método en la función)

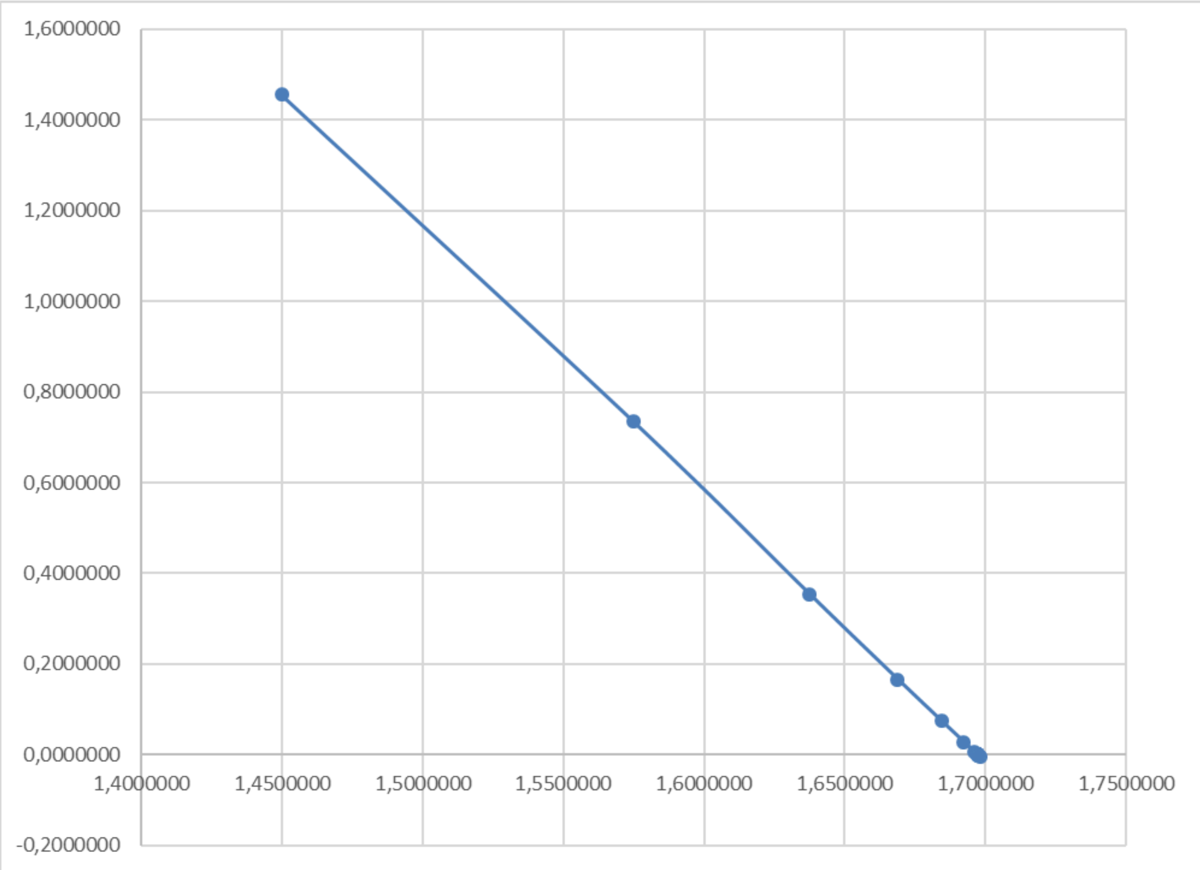
Para la segunda raíz los valores de 1.2 y 2.2




```
 - B I S E C C I O N -
Digite el extremo inferior (a) : 1.2
Digite el extremo superior (b) : 2.2
Digite el numero de iteraciones : 200
```

n	Raiz	f(Raiz)	Er
2	1.4500000000	1.455203343	0.1724137931
3	1.5750000000	0.7341235205	0.07936507937
4	1.6375000000	0.3539154125	0.03816793893
5	1.6687500000	0.1662556004	0.01872659176
6	1.6843750000	0.07401074157	0.009276437848
7	1.6921875000	0.0284081056	0.004616805171
8	1.6960937500	0.005752235862	0.002303086135
9	1.6980468750	-0.00553739751	0.001150218542
10	1.6970703125	0.0001041696482	0.0005754402118
11	1.6975585938	-0.002717430643	0.0002876373468
12	1.6973144531	-0.001306834134	0.0001438393602
13	1.6971923828	-0.0006013830846	7.192485291e-05
14	1.6971313477	-0.0002486194201	3.59637198e-05
15	1.6971008301	-7.222806039e-05	1.798218325e-05
16	1.6970855713	1.597000043e-05	8.991172467e-06
17	1.6970932007	-2.812922837e-05	4.495566023e-06
18	1.6970893860	-6.07966356e-06	2.247788064e-06
19	1.6970874786	4.945156037e-06	1.123895295e-06
20	1.6970884323	-5.672568615e-07	5.619473318e-07
21	1.6970879555	2.188948813e-06	2.809737449e-07
22	1.6970881939	8.10845781e-07	1.404868527e-07
23	1.6970883131	1.217944112e-07	7.024342141e-08

Como podemos observar, para la segunda raíz positiva, el método de Bisección requiere un total de **22 iteraciones** para aproximarse a la solución. Esto refleja nuevamente que, si bien el método es confiable y garantiza la convergencia, no es la opción más eficiente en términos de uso de recursos computacionales.



(gráfica con los datos de desempeño del método en la función)

MÉTODO 2

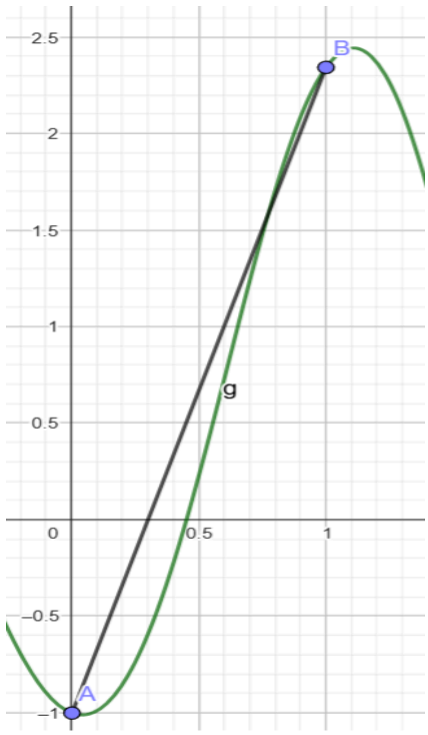
REGLA FALSA

Para este método, se traza una recta que une dos puntos, y al igual que en el método de Bisección, es necesario que se cumpla la siguiente condición:

- Debe haber un cambio de signo en xi y xs, es decir, f(xi)·f(xs)<0, lo que garantiza la existencia de una raíz en el intervalo.

Dado que tanto la Regla Falsa como la Bisección comparten este requisito, y con el objetivo de mantener la consistencia en el análisis, se decidió utilizar los mismos puntos iniciales que en el método de Bisección. Esto permite realizar una comparación más equitativa entre ambos métodos y evaluar sus diferencias en términos de eficiencia y precisión.

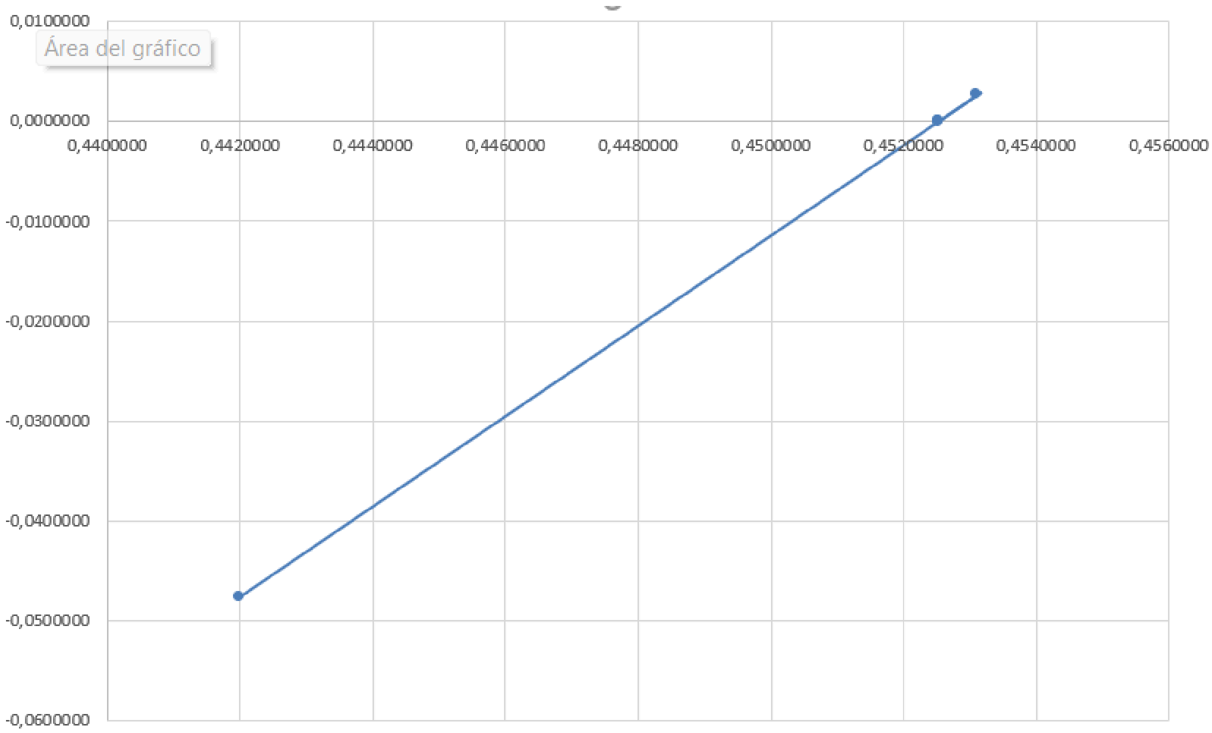
Para la primera raíz positiva:




```
 - R E G L A   F A L S A -
Digite el extremo inferior (Xi) : 0
Digite el extremo superior (Xs) : 1
Digite el numero de iteraciones : 200

 XR : 0.2992338013
n       Raiz                    f(Raiz)                 Er
1       0.4419953974            -0.04759207378          0.3229933999
2       0.4531094526            0.002634866741          0.02452841174
3       0.4525264178            -2.1186224e-05          0.001288399487
4       0.4525310684            -9.113972188e-09                1.027691841e-05
5       0.4525310704            -3.920613834e-12                4.420949368e-09
Procedimiento completado satisfactoriamente
```

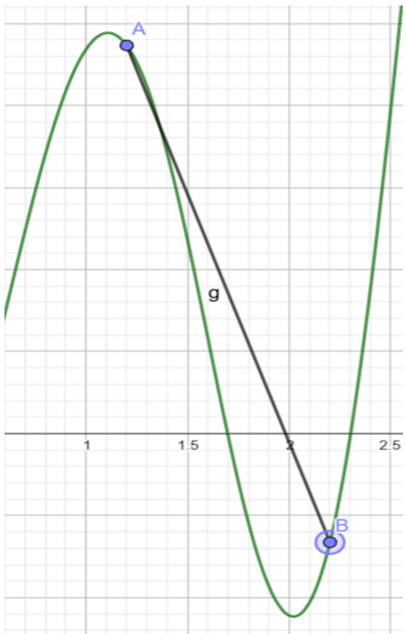
Como podemos observar, este método requiere sólo **5 iteraciones**, lo que lo hace mucho más eficiente en comparación con la Bisección, tanto en términos de iteraciones como en la cota de error, que resulta más precisa.

El buen desempeño de la Regla Falsa en este caso podría deberse a que, cerca de la raíz, la función se comporta de manera casi lineal, lo que permite que las aproximaciones mejoren rápidamente en cada iteración.



(gráfica con los datos de desempeño del método en la función)

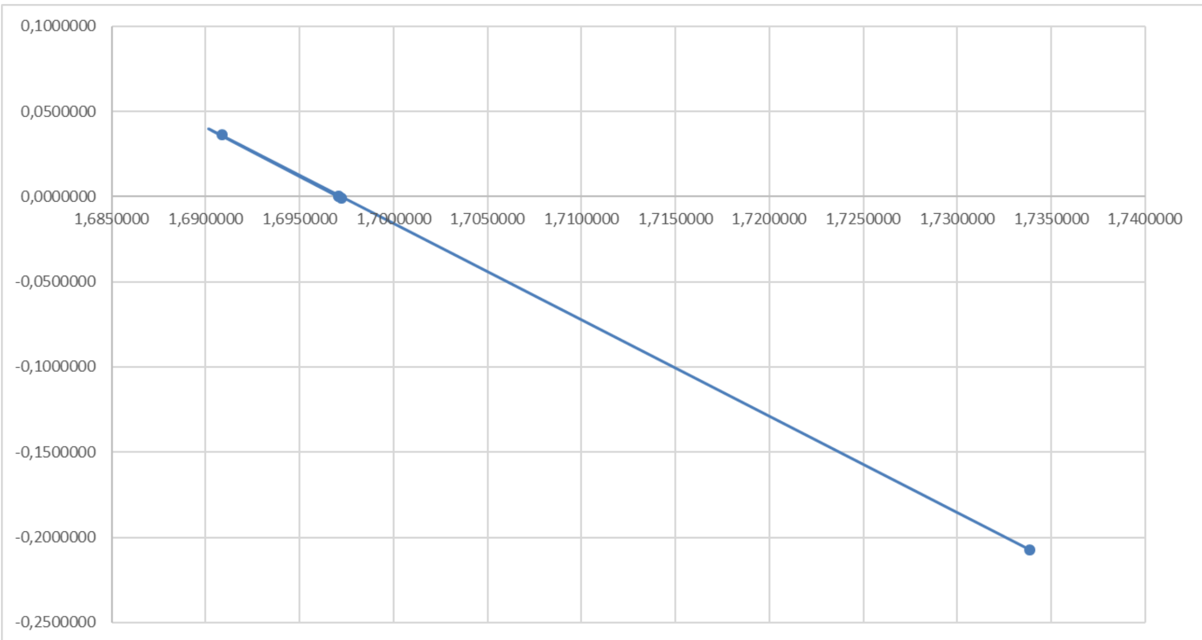
Para la segunda raíz positiva



```
 - R E G L A   F A L S A -
Digite el extremo inferior (Xi) : 1.2
Digite el extremo superior (Xs) : 2.2
Digite el numero de iteraciones : 200

 XR : 1.98043157
n       Raiz                    f(Raiz)                 Er
1       1.7338349581            -0.2071851231           0.1422261161
2       1.6908542301            0.03616369844           0.02541953485
3       1.6972415307            -0.0008854238306                0.003763342188
4       1.6970888829            -3.171712995e-06                8.994686968e-05
5       1.6970883361            -1.135305427e-08                3.221742164e-07
6       1.6970883342            -4.063926973e-11                1.153212826e-09
Procedimiento completado satisfactoriamente
```

Aquí también se observa una **mejora considerable** en comparación con el método de Bisección. En este caso, la Regla Falsa **requiere solo 6 iteraciones** para aproximarse a la raíz, lo que representa una reducción significativa en el número de pasos necesarios.

Además, se evidencia una **mejoría en la cota de error**, lo que indica que el método no solo es más eficiente en términos de iteraciones, sino que también proporciona una aproximación más precisa en menos tiempo.



(gráfica con los datos de desempeño del método en la función)

MÉTODO 3

SECANTE

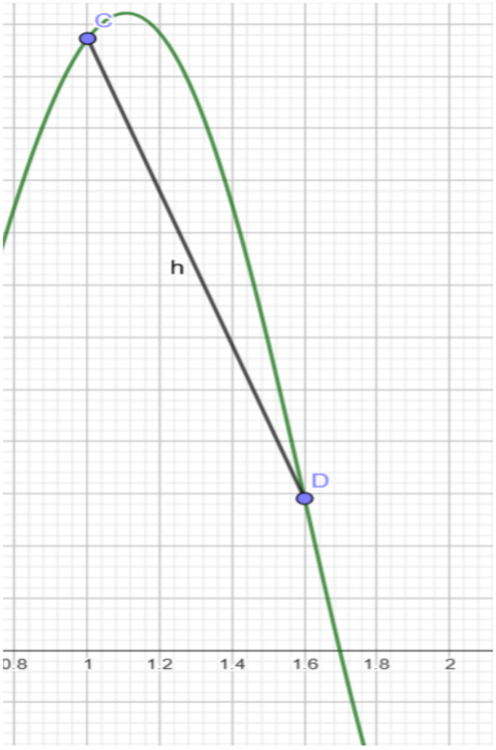
Para la primera raíz, la elección del intervalo en el método de la Secante fue diferente en comparación con los métodos cerrados (Bisección y Regla Falsa). Dado que la Secante no requiere un intervalo con cambio de signo, la selección de los puntos iniciales influye directamente en su rendimiento.

Por lo que se seleccionó el intervalo [0,0.4], ya que ofrece buenos resultados al acercarse rápidamente a la raíz. Además de que se encuentra dentro del intervalo 0-1 lo que permite una mejor simetría a la hora de comparar los métodos .

```
 - S E C A N T E -
Digite Xo : 0
Digite X1 : 0.4
Digite el numero de iteraciones : 200

 XR : 0.5188163026
n       Raiz                    f(raiz)                 Er
1       0.4500238196                    -0.01139565554          0.1528640931
2       0.4524260629                    -0.0004781271544                0.005309692606
3       0.4525312678                    8.988078922e-07         0.000232481009
4       0.4525310704                    -7.031683569e-11                4.362098785e-07
5       0.4525310704                    -5.551115123e-17                3.412358186e-11
Procedimiento completado satisfactoriamente
```

Como podemos observar, el método de la Secante es más eficiente en términos de iteraciones, ya que requiere solo **5 iteraciones** para aproximarse a la raíz, superando a los demás métodos en este aspecto.A la vez su cota de error es menor que en los casos anteriores.

Para la segunda raíz positiva, utilizamos los puntos 1 y 1.6 como valores iniciales. En este caso, también trabajamos con un intervalo más pequeño en comparación con los utilizados anteriormente.




```
 - S E C A N T E -
Digite Xo : 1
Digite X1 : 1.6
Digite el numero de iteraciones : 200

 XR : 1.798437794
n       Raiz                    f(raiz)                 Er
1       1.7031233640            -0.03475671463          0.05596448972
2       1.6965398664            0.00317126648           0.003880544043
3       1.6970903314            -1.154405409e-05                0.0003243580929
4       1.6970883348            -3.724517139e-09                1.176448215e-06
5       1.6970883342            3.552713679e-15         3.796860623e-10
Procedimiento completado satisfactoriamente
```

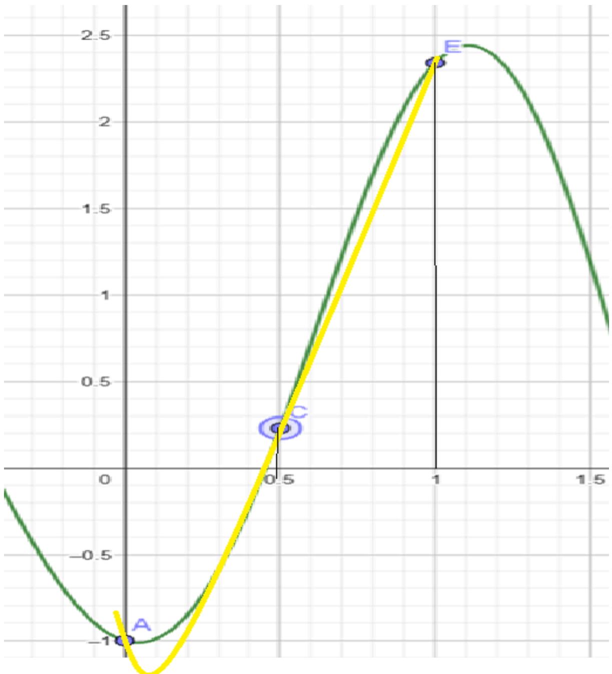
El método de la Secante para la segunda raíz positiva muestra resultados altamente eficientes, logrando la aproximación en solo **5 iteraciones.** Además, ofrece una cota de error más precisa en comparación con la Regla Falsa y la Bisección, lo que indica una mejor convergencia en este caso.

MÉTODO 4

MULLER

Para la primera raíz positiva, utilizaremos los puntos x0,x1,x2 necesarios para aplicar el método de Müller.

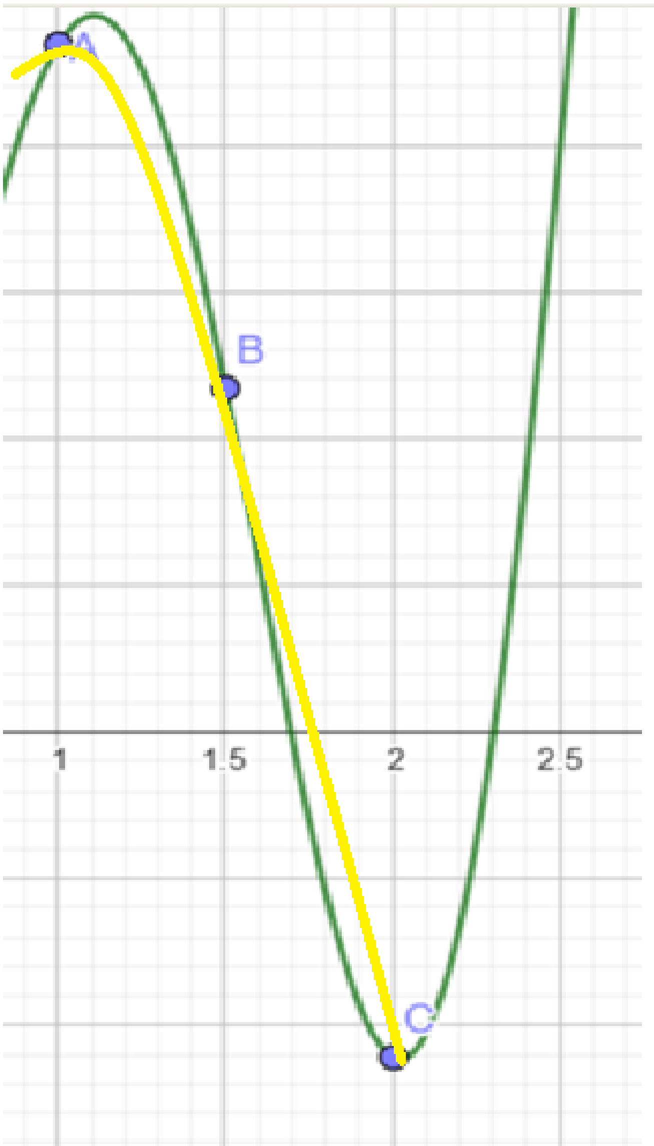
En este caso, seleccionamos los valores **x0=0, x1=0.5 5 y x2=1,** los cuales forman una parábola que pasa por estos puntos, cumpliendo con los requisitos del método,no utilizaremos valores más alejados debido a la cercanía de las raíces entre sí . Esta elección permite que la interpolación cuadrática se ajuste de manera eficiente a la función, facilitando una convergencia rápida hacia la raíz.




```
 - M U L L E R -
Digite el valor X0 : 0
Digite el valor X1 : 0.5
Digite el valor X2 : 1
Digite el numero de iteraciones : 200

n       Raiz                            f(raiz)                 COTA DE ERROR
2       1.0000000000                    2.341868451             1.322070298
3       0.4306501835                    -0.09795197253          0.04650524001
4       0.4516544836                    -0.003989033922         0.001973719236
5       0.4525476857                    7.566060386e-05         3.671541263e-05
6       0.4525310709                    1.868016936e-09         9.065160665e-10
procedimiento completado satisfactoriamente
```

En el método de Müller, la raíz se encuentra en solo 5 iteraciones, lo que demuestra una buena eficiencia en comparación con los otros métodos analizados.Además, su cota de error es mejor que en otros casos anteriores, lo que indica una mayor precisión en la aproximación de la raíz.

Para la segunda raíz positiva utilizaremos los puntos x0=1,x1=1.5,x2=2



```
 - M U L L E R -
Digite el valor X0 : 1
Digite el valor X1 : 1.5
Digite el valor X2 : 2
Digite el numero de iteraciones : 200

n       Raiz                            f(raiz)                 COTA DE ERROR
2       2.0000000000                    -1.108843916            0.1187865547
3       1.7876510864                    -0.4863370919           0.06099970579
4       1.6848742527                    0.07108544137           0.006819071279
5       1.6964424145                    0.003734951545          0.0003815071246
6       1.6970898663                    -8.856196515e-06                9.028058844e-07
7       1.6970883342                    -1.498461355e-10                1.527566432e-11
procedimiento completado satisfactoriamente
```

En este caso,aunque requiere **6 iteraciones**, supera a los demás métodos en precisión, ya que su cota de error es significativamente mejor que en los casos anteriores.

MÉTODO	**VALOR(ES) INICIAL(ES)**	**RAÍZ**	**ITERACIONES**	**COTA DE ERROR**
BISECCIÓN	**Xa**=0-**Xb**=1	0,4525310695	24	6.585696407e-08
	Xa=1.2-**Xb**=2.2	1,6970883131	22	7.024342141e-08
REGLA FALSA	[0-1]	0,4525310704	5	4.420949368e-09
	[1.2-2.2]	1,6970883361	6	1.153212826e-09
SECANTE	[0-0.4]	0.452310704	5	3.412358186e-11
	[1-1.6]	1.6970883342	5	3.796860623e-10
MULLER	**X0=**0-**X1=**0.5-**X2**=1	0.4525310709	5	9.065160665e-10
	X0=1-**X1=**1.5-**X2**=2	1.6970883342	6	1.527566432e-11

Tras aplicar los métodos de Bisección, Regla Falsa, Secante y Müller, se observa un comportamiento distintivo en cada uno. Bisección es el método más lento, requiriendo 24 y 22 iteraciones para encontrar las raíces. Aunque garantiza convergencia, su cota de error es mayor en comparación con otros métodos, lo que lo hace menos eficiente.

Regla Falsa mejora la eficiencia al reducir considerablemente las iteraciones a 5 y 6. Su desempeño depende de la forma de la función, y en este caso, su comportamiento fue favorable, ya que cerca de la raíz la función es casi lineal. No obstante, este método puede estancarse si la recta secante no representa bien la curva en el intervalo. A pesar de ello, ofrece una mejor cota de error que Bisección, mostrando una ventaja en precisión y rapidez.

El método Secante, con la nueva elección de intervalos, sigue siendo eficiente en términos de iteraciones, tomando 5 pasos para ambas raíces. Sin embargo, su cota de error es ahora la mejor de todos los métodos probados en la primera raíz, con 3.412358186e-11, lo que indica que, en este caso particular, logró gran precisión sin necesidad de muchas iteraciones. Esto muestra que la elección de los valores iniciales influye significativamente en su desempeño.

Finalmente, Müller mantiene un buen equilibrio entre eficiencia y precisión. Aunque requiere 5 y 6 iteraciones, su cota de error es la más baja en la segunda raíz con 1.527566432e-11, superando a los demás métodos. Su uso de interpolación cuadrática le permite adaptarse mejor a la curvatura de la función, lo que explica su mayor precisión con un número moderado de iteraciones.