

Contenido

% METODO DE LA SECANTE	2
METODO DE LA SECANTE TEST	4
% REGLA FALSA	6
REGLA FALSA TEST.....	8
% METODO DE BUSQUEDAS INCREMENTALES	10
METODO DE BUSQUEDAS INCREMENTALES TEST	12
%METODO DE BISECCION	14
METODO DE BISECCION TEST	16
%METODO DE PUNTO FIJO	19
METODO DE PUNTO FIJO TEST	21

% METODO DE LA SECANTE

```
fprintf('METODO DE LA SECANTE\n\n\n');
```

format long;%format long permite utilizar la maxima capacidad del computador

```
Xo = input('ingrese xo\n');  
X1 = input('\ningrese x1\n');  
Tol = input('\ningrese la tolerancia\n');  
Iter = input('\ningrese el numero de iteraciones\n');  
Fun = input('\ningrese la funcion entre comillas simples\n');
```

f=inline (Fun); %El comando inline permite hacer la asignacion posterior de variables en una funcion

```
yo=f(Xo);
```

```
if yo == 0  
    fprintf('\n\nSOLUCION:\n')  
    fprintf('xo es raiz\n');
```

```
else  
    y1 =f(X1);  
    d =(y1-yo);  
    e =Tol+1;  
    cont =0;  
    Z1 = [cont,X1, y1, e];
```

```

Z = [cont,X1, y1, e];
%Z es una matriz la cual permitira observar lo datos como una tabla al final del programa

while y1~=0 & e>Tol && cont<Iter && d~=0

    X2 = X1-((y1*(X1-Xo))/(d));
    e = abs((X2-X1)/X2);
    %e = abs(X2-X1);
    Xo = X1;
    yo = y1;
    y1 = f(X2);
    X1 = X2;
    d = (y1-yo);
    cont = cont+1;
    Z(cont,1) = cont;
    Z(cont,2) = X1;
    Z(cont,3) = y1;
    Z(cont,4) = e;
    %las z son las posiciones asignadas en la tabla a los resultados que se observaron

end

if y1 == 0
    fprintf('\n\nSOLUCION:\n')
    fprintf('%g es raiz\n\n',X1);
else
    if e < Tol
        fprintf('\n\nSOLUCION:\n')
        fprintf(' %g es una aproximacion a una raiz con una tolerancia %g \n\n',X1,Tol)
    else
        if d == 0
            fprintf('\n\nSOLUCION:\n')
            fprintf('el denominador es cero, FRACASO\n\n');
        else
            fprintf('\n\nSOLUCION:\n')
            fprintf('Fracaso en %g iteraciones\n\n',Iter);
        end
    end
end
end
end
fprintf('TABLA\n\ninteraciones      Xn      y1      Error relativo\n\n');
disp(Z1);
disp(Z);
ezplot(f);
%fplot(f,[-1 15]); %Muestra la funcion graficada

grid on %Muestra cuadrícula en la grafica de la funcion

```

METODO DE LA SECANTE TEST

METODO DE LA SECANTE

ingrese xo

0

ingrese x1

2

ingrese la tolerancia

0.00000000001

ingrese el numero de iteraciones

100

ingrese la funcion entre comillas simples

'x^2-3'

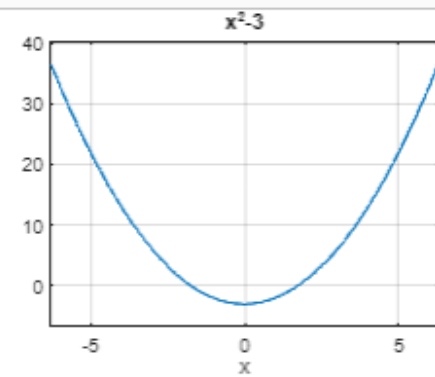
SOLUCION:

1.73205 es una aproximacion a una raiz con una tolerancia 1e-11

TABLA

iteraciones	Xn	y1	Error relativo
0	2.0000000000000000	1.0000000000000000	1.000000000010000
1.0000000000000000	1.5000000000000000	-0.7500000000000000	0.3333333333333333
2.0000000000000000	1.714285714285714	-0.061224489795919	0.1250000000000000
3.0000000000000000	1.7333333333333333	0.0044444444444445	0.010989010989011
4.0000000000000000	1.732044198895028	-0.000022893074082	0.000744284954811
5.0000000000000000	1.732050805123027	-0.000000008472674	0.000003814107519
6.0000000000000000	1.732050807568882	0.000000000000016	0.00000001412115
7.0000000000000000	1.732050807568877	0.000000000000000	0.000000000000003

Figure 1 x +



Command Window

SOLUCION:

1.73205 es una aproximacion a una raiz con una tolerancia 1e-11

TABLA

iteraciones	x_n	y_1	Error relativo
0	2.000000000000000	1.000000000000000	1.00000000010000
1.000000000000000	1.500000000000000	-0.750000000000000	0.333333333333333
2.000000000000000	1.714285714285714	-0.061224489795919	0.125000000000000
3.000000000000000	1.733333333333333	0.004444444444445	0.010989010989011
4.000000000000000	1.732044198895028	-0.000022893074082	0.000744284954811
5.000000000000000	1.732050805123027	-0.000000008472674	0.00003814107519
6.000000000000000	1.732050807568882	0.000000000000016	0.00000001412115
7.000000000000000	1.732050807568877	0.000000000000000	0.000000000000003

%REGLA FALSA

```
fprintf('METODO REGLA FALSA\n\n\n');
format long;%format long permite utilizar la maxima capacidad del computador
```

```
Xi = input ('Ingrese el limite inferior del intervalo\n');
Xs = input ('\nIngrese el limite superior del intervalo\n');
Tol = input ('\nIngrese la tolerancia deseada\n');
Iter = input ('\nIngrese el numero de iteraciones\n');
Fun = input ('\nIngrese la funcion entre comillas simples\n');
```

f = inline (Fun);%El comando inline permite hacer la asignacion posterior de variables en una funcion.

$$Y_i = f(X_i);$$
$$Y_S = f(X_S);$$

```

if Yi == 0
    fprintf('\n\nSOLUCION:\n')
    fprintf('Xi es raiz\n\n');
else
    if Ys == 0
        fprintf('\n\nSOLUCION:\n')
        fprintf('Xs es raiz\n\n');
    else
        if Yi*Ys < 0
            Xm = (Xi)-((f(Xi)*(Xi-Xs))/(f(Xi)-f(Xs)));
            Ym = f(Xm);
            Error = Tol+1;
            Cont = 1;
            Z = [Cont,Xi,Xs,Xm,Ym,Error];
            %Z es una matriz la cual permitira observar lo datos como una tabla al final del programa

```

```
while Ym~=0 && Error>Tol && Cont<Iter
```

```

if Yi*Ym < 0
    Xs = Xm;
    Ys = Ym;
else
    Xi = Xm;
    Yi = Ym;
end
Xaux = Xm;
Xm = (Xi)-((f(Xi)*(Xi-Xs))/(f(Xi)-f(Xs)));
Ym = f(Xm);

```

```

Error = abs(Xm-Xaux)/Xm;
Cont = Cont+1;
Z(Cont,1) = Cont;
Z(Cont,2) = Xi;
Z(Cont,3) = Xs;
Z(Cont,4) = Xm;
Z(Cont,5) = Ym;
Z(Cont,6) = Error;
%las z son las posiciones asignadas en la tabla a los resultados que se observaron

end
if Ym == 0
    fprintf('\n\nSOLUCION:\n')
    fprintf('%g es raiz\n\n',Xm);
else
    if Error<Tol
        fprintf('\n\nSOLUCION:\n')
        fprintf(' %g es una aproximacion a una raiz con una tolerancia %g \n\n',Xm,Tol);
    else
        fprintf('\n\nSOLUCION:\n')
        fprintf('Fracaso en %g iteraciones\n\n',Iter);
    end
end
else
    fprintf('\n\nSOLUCION:\n')
    fprintf('El intervalo es inadecuado\n\n');
end
end
end
fprintf('TABLA\n\nIteraciones      Xi      Xs      Xm      Ym      Error
Absoluto\n\n');
disp(Z)
ezplot(f);
%fplot(f,[-1 10]);%Muestra la funcion graficada
grid on %Muestra cuadrícula en la grafica de la función

```

REGLA FALSA TEST

METODO REGLA FALSA

Ingrese el limite inferior del intervalo

0

Ingrese el limite superior del intervalo

2

Ingrese la tolerancia deseada

0.000000005

Ingrese el numero de iteraciones

100

Ingrese la funcion entre comillas simples

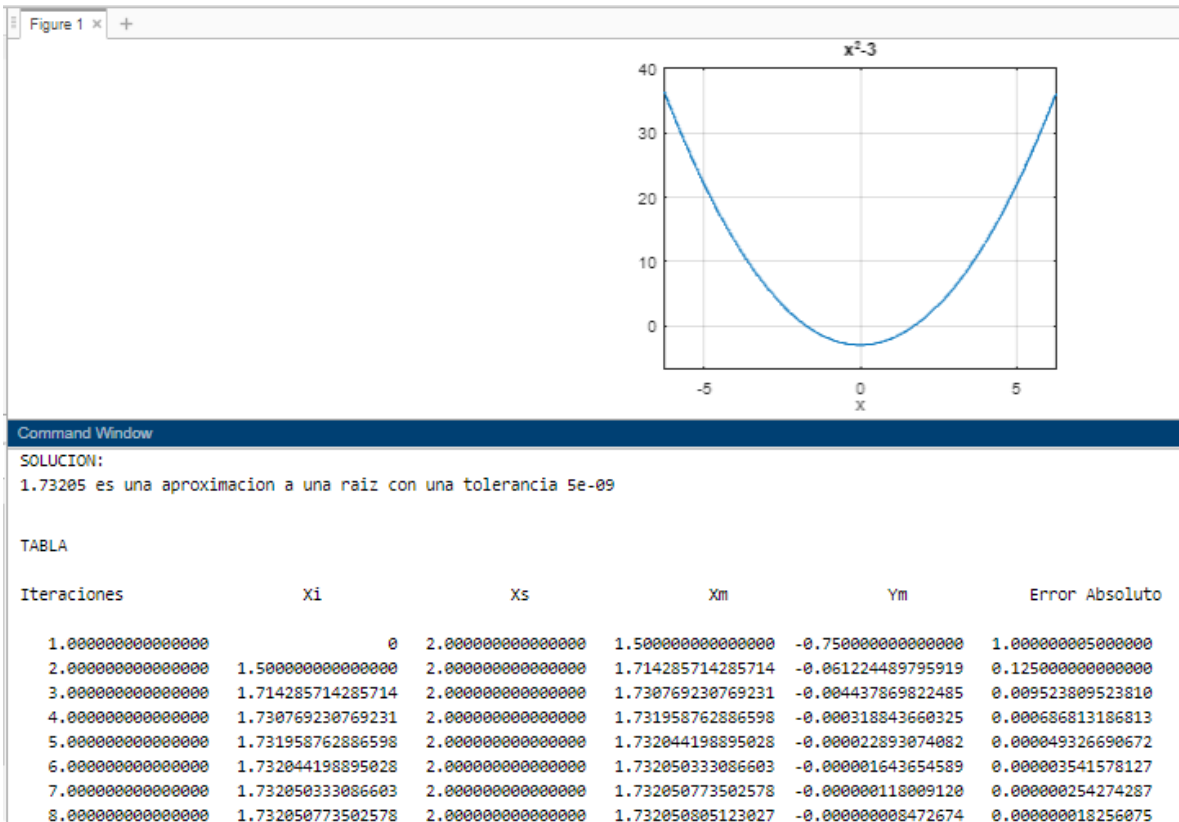
'x^2-3'

SOLUCION:

1.73205 es una aproximacion a una raiz con una tolerancia 5e-09

TABLA

Iteraciones	X_i	X_s	X_m	Y_m	Error Absoluto
1.0000000000000000	0	2.0000000000000000	1.5000000000000000	-	
0.7500000000000000	1.0000000050000000				
2.0000000000000000	1.5000000000000000	2.0000000000000000	1.714285714285714	-	
0.061224489795919	0.1250000000000000				
3.0000000000000000	1.714285714285714	2.0000000000000000	1.730769230769231	-	
0.004437869822485	0.009523809523810				
4.0000000000000000	1.730769230769231	2.0000000000000000	1.731958762886598	-	
0.000318843660325	0.000686813186813				
5.0000000000000000	1.731958762886598	2.0000000000000000	1.732044198895028	-	
0.000022893074082	0.000049326690672				
6.0000000000000000	1.732044198895028	2.0000000000000000	1.732050333086603	-	
0.000001643654589	0.000003541578127				
7.0000000000000000	1.732050333086603	2.0000000000000000	1.732050773502578	-	
0.000000118009120	0.000000254274287				
8.0000000000000000	1.732050773502578	2.0000000000000000	1.732050805123027	-	
0.000000008472674	0.000000018256075				
9.0000000000000000	1.732050805123027	2.0000000000000000	1.732050807393273	-	
0.000000000608310	0.000000001310727				



% METODO DE BUSQUEDAS INCREMENTALES

format long %format long permite utilizar la maxima capacidad del computador

```
fprintf('BUSQUEDAS INCREMENTALES\n\n\n')
```

```
Xo = input('Ingrese el valor inicial\n');  
Delta = input('\nIngrese el incremento (Delta)\n');  
Iter = input ('\nIngrese el numero de iteraciones\n');  
Fun = input ('\nIngrese la funcion en comillas simples\n');
```

f = inline(Fun); %El comando inline permite hacer la asignacion posterior de variables en una funcion.

```
Yo = f(Xo);
```

```
if Yo==0  
    fprintf('\n\nSOLUCION:\n')  
    fprintf ('Xo es raiz\n\n');
```

```
else
```

```
    X1 = Xo+Delta;
```

```
    Cont = 1;
```

```
    Y1 = f(X1);
```

```
    Z = [Cont,Xo,Yo,Yo*Y1];
```

%Z es una matriz la cual permitira observar lo datos como una tabla al final del programa

```
while Yo*Y1>0 & Cont<Iter
```

```
    Xo = X1;
```

```
    Yo = Y1;
```

```
    X1 = Xo+Delta;
```

```
    Y1 = f(X1);
```

```
    Cont = Cont+1;
```

```
    Z(Cont,1) = Cont;
```

```
    Z(Cont,2) = Xo;
```

```
    Z(Cont,3) = Yo;
```

```
    Z(Cont,4) = Yo*Y1;
```

%las z son las posiciones asignadas en la tabla a los resultados que se observaron

```
end
```

```
if Y1==0
```

```
    fprintf('\n\nSOLUCION:\n')
```

```
    fprintf('%g es raiz\n\n',X1);
```

```
else
```

```
    if Yo*Y1<0
```

```
        fprintf('\n\nSOLUCION:\n')
```

```
        fprintf ('El intervalo que contiene la raiz es[%g,%g]\n\n',Xo,X1);
```

```

    else
        fprintf('\n\nSOLUCION:\n')
        fprintf ('Fracaso en %g iteraciones\n\n',Iter);
    end
end
end
fprintf('TABLA\n\nIteraciones      Xo      Yo      Yo*Y1 \n\n');
disp(Z); %Muestra los datos en forma de tabla
ezplot(f); %Muestra la funcion graficada
grid on %Muestra cuadrícula en la grafica de la funcion

```

METODO DE BUSQUEDAS INCREMENTALES TEST

BUSQUEDAS INCREMENTALES

Ingrese el valor inicial

1.5

Ingrese el incremento (Delta)

0.05

Ingrese el numero de iteraciones

100

Ingrese la funcion en comillas simples

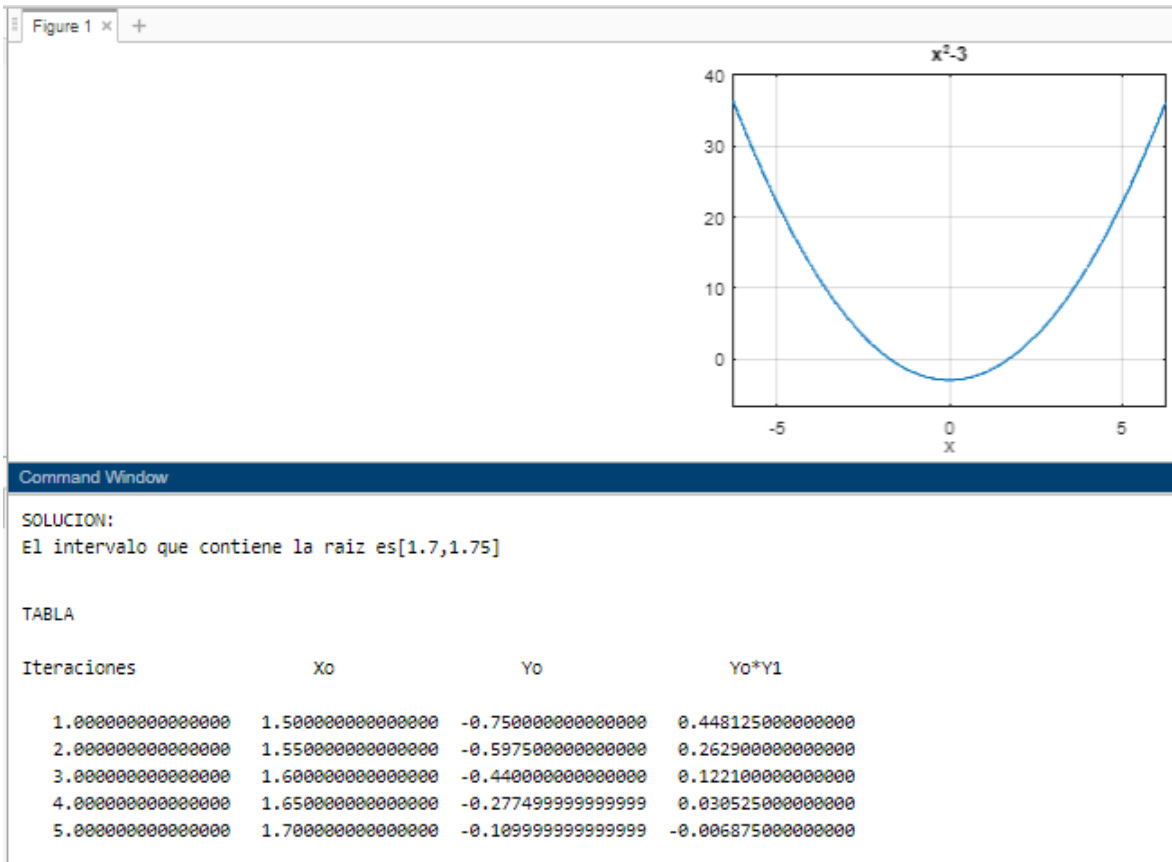
'x^2-3'

SOLUCION:

El intervalo que contiene la raiz es[1.7,1.75]

TABLA

Iteraciones	Xo	Yo	Yo*Y1
1.0000000000000000	1.5000000000000000	-0.7500000000000000	0.4481250000000000
2.0000000000000000	1.5500000000000000	-0.5975000000000000	0.2629000000000000
3.0000000000000000	1.6000000000000000	-0.4400000000000000	0.1221000000000000
4.0000000000000000	1.6500000000000000	-0.2774999999999999	0.0305250000000000
5.0000000000000000	1.7000000000000000	-0.1099999999999999	-0.0068750000000000



%METODO DE BISECCION

```
fprintf('METODO DE BISECCION\n\n');
```

format long%format long permite utilizar la maxima capacidad del computador

```
Xi=input('Ingrese el limite inferior del intervalo\n');
Xs=input('\nIngrese el limite superior del intervalo\n');
Tol=input('\nIngrese la tolerancia deseada\n');
Iter=input('\nIngrese el numero de iteraciones\n');
Fun=input('\nIngrese la funcion entre comillas simples\n');
```

f = inline (Fun);%El comando inline permite hacer la asignacion posterior de variables en una funcion.

```
Yi = f(Xi);
Ys = f(Xs);
```

```
if Yi == 0
    fprintf('\n\nSOLUCION:\n')
    fprintf('Xi es raiz\n\n');
```

```
else
    if Ys == 0
        fprintf('\n\nSOLUCION:\n')
        fprintf('Xs es raiz\n\n');
```

```
    else
        if Yi*Ys < 0
            Xm = (Xi+Xs)/2;
            Ym = f(Xm);
            Error = Tol+1;
            Cont = 1;
            Z = [Cont,Xi,Xs,Xm,Ym,Yi*Ym,Ym*Ys>Error];
```

%Z es una matriz la cual permitira observar lo datos como una tabla al final del programa

```
while Ym~=0 && Error>Tol && Cont<Iter
```

```
    if Yi*Ym<0
        Xs=Xm;
        Ys=Ym;
    else
        Xi=Xm;
        Yi=Ym;
    end
    Xaux = Xm;
    Xm = (Xi+Xs)/2;
    Ym = f(Xm);
    Error = abs(Xm-Xaux);
```

```

Cont = Cont+1;
Z(Cont,1) = Cont;
Z(Cont,2) = Xi;
Z(Cont,3) = Xs;
Z(Cont,4) = Xm;
Z(Cont,5) = Ym;
Z(Cont,6) = Yi*Ym;
Z(Cont,7) = Ym*Ys;
Z(Cont,8) = Error;

```

%las z son las posiciones asignadas en la tabla a los resultados que se observaron

```

end
if Ym == 0
    fprintf('\n\nSOLUCION:\n')
    fprintf('%g es raiz\n\n',Xm);
else
    if Error<Tol
        fprintf('\n\nSOLUCION:\n')
        fprintf(' %g es una aproximacion a una raiz con una tolerancia %g \n\n',Xm,Tol);
    else
        fprintf('\n\nSOLUCION:\n')
        fprintf('Fracaso en %g iteraciones\n\n',Iter);
    end
end
else
    fprintf('\n\nSOLUCION:\n')
    fprintf('El intervalo es inadecuado\n\n');
end
end
end
fprintf('TABLA\n\nIteraciones      Xi      Xs      Xm      Ym      Yi*Ym\n\nYm*Ys      Error Absoluto\n\n');

disp(Z);

ezplot(f); %Muestra la funcion graficada
grid on %muestra cuadrícula en la grafica de la funcion

```

METODO DE BISECCION TEST

METODO DE BISECCION

Ingrese el limite inferior del intervalo

0

Ingrese el limite superior del intervalo

2

Ingrese la tolerancia deseada

0.0000001

Ingrese el numero de iteraciones

100

Ingrese la funcion entre comillas simples

'x^2-3'

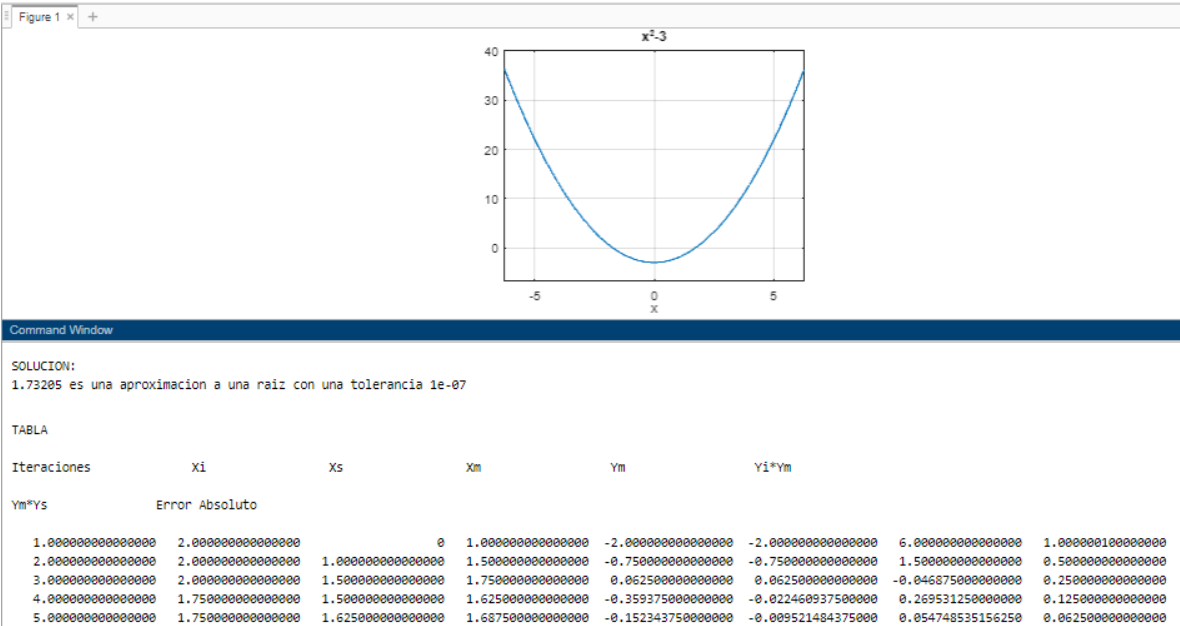
SOLUCION:

1.73205 es una aproximacion a una raiz con una tolerancia 1e-07

TABLA

Iteraciones	Xi	Xs	Xm	Ym	Yi*Ym
Ym*Ys	Error Absoluto				
1.0000000000000000	0	2.0000000000000000	1.0000000000000000	-	
2.0000000000000000	6.0000000000000000	-2.0000000000000000	1.0000001000000000		
2.0000000000000000	1.0000000000000000	2.0000000000000000	1.5000000000000000	-	
0.7500000000000000	1.5000000000000000	-0.7500000000000000	0.5000000000000000		
3.0000000000000000	1.5000000000000000	2.0000000000000000	1.7500000000000000		
0.0625000000000000	-0.0468750000000000	0.0625000000000000	0.2500000000000000		
4.0000000000000000	1.5000000000000000	1.7500000000000000	1.6250000000000000	-	
0.3593750000000000	0.2695312500000000	-0.0224609375000000	0.1250000000000000		
5.0000000000000000	1.6250000000000000	1.7500000000000000	1.6875000000000000	-	
0.1523437500000000	0.0547485351562500	-0.0095214843750000	0.0625000000000000		
6.0000000000000000	1.6875000000000000	1.7500000000000000	1.7187500000000000	-	
0.0458984375000000	0.0069923400878910	-0.0028686523437500	0.0312500000000000		
7.0000000000000000	1.7187500000000000	1.7500000000000000	1.7343750000000000		
0.0080566406250000	-0.0003697872161870	0.0005035400390620	0.0156250000000000		

8.000000000000000 1.718750000000000 1.734375000000000 1.726562500000000 -
0.018981933593750 0.000871241092682 -0.000152930617332 0.007812500000000
9.000000000000000 1.726562500000000 1.734375000000000 1.730468750000000 -
0.005477905273438 0.000103981234133 -0.000044133514166 0.003906250000000
10.000000000000000 1.730468750000000 1.734375000000000 1.732421875000000
0.001285552978516 -0.000007042137440 0.000010357238352 0.001953125000000
11.000000000000000 1.730468750000000 1.732421875000000 1.731445312500000 -
0.002097129821777 0.000011487878510 -0.000002695971489 0.000976562500000
12.000000000000000 1.731445312500000 1.732421875000000 1.731933593750000 -
0.000406026840210 0.000000851490995 -0.000000521969014 0.000488281250000
13.000000000000000 1.731933593750000 1.732421875000000 1.732177734375000
0.000439703464508 -0.000000178531408 0.000000565262098 0.000244140625000
14.000000000000000 1.731933593750000 1.732177734375000 1.732055664062500
0.000016823410988 -0.000000006830756 0.000000007397312 0.000122070312500
15.000000000000000 1.731933593750000 1.732055664062500 1.731994628906250 -
0.000194605439901 0.000000079015032 -0.000000003273927 0.000061035156250
16.000000000000000 1.731994628906250 1.732055664062500 1.732025146484375 -
0.000088891945779 0.000000017298856 -0.000000001495466 0.000030517578125
17.000000000000000 1.732025146484375 1.732055664062500 1.732040405273438 -
0.000036034500226 0.000000003203177 -0.000000000606223 0.000015258789062
18.000000000000000 1.732040405273438 1.732055664062500 1.732048034667969 -
0.000009605602827 0.000000000346133 -0.000000000161599 0.000007629394531
19.000000000000000 1.732048034667969 1.732055664062500 1.732051849365234
0.000003608889529 -0.000000000034666 0.000000000060714 0.000003814697266
20.000000000000000 1.732048034667969 1.732051849365234 1.732049942016602 -
0.000002998360287 0.000000000028801 -0.000000000010821 0.000001907348633
21.000000000000000 1.732049942016602 1.732051849365234 1.732050895690918
0.000000305263711 -0.000000000000915 0.000000000001102 0.000000953674316
22.000000000000000 1.732049942016602 1.732050895690918 1.732050418853760 -
0.000001346548515 0.000000000004037 -0.000000000000411 0.000000476837158
23.000000000000000 1.732050418853760 1.732050895690918 1.732050657272339 -
0.000000520642459 0.000000000000701 -0.000000000000159 0.000000238418579
24.000000000000000 1.732050657272339 1.732050895690918 1.732050776481628 -
0.000000107689388 0.000000000000056 -0.000000000000033 0.000000119209290
25.000000000000000 1.732050776481628 1.732050895690918 1.732050836086273
0.000000098787158 -0.000000000000011 0.000000000000030 0.000000059604645



%METODO DE PUNTO FIJO

```
fprintf('METODO PUNTO FIJO\n\n');
```

format long;%format long permite utilizar la maxima capacidad del computador

```
Xo = input('ingrese el valor inicial\n');
```

```
Iter = input('\ningrese el numero de iteraciones\n');
```

```
Tol = input('\ningrese la tolerancia que desea\n');
```

```
Fun = input('\ningrese la funcion en comillas simples\n');
```

```
G = input('\ningrese la funcion despejada en comillas simples\n');
```

```
f = inline(Fun);
```

g = inline(G);%El comando inline permite hacer la asignacion posterior de variables en una funcion.

```
Yn = f(Xo);
```

```
Error = Tol+1;
```

```
Cont = 0;
```

```
Z1 = [Cont,Xo,Yn>Error];
```

%Z es una matriz la cual permitira observar lo datos como una tabla al final del programa

```
Z = [Cont,Xo,Yn>Error];
```

```
while Yn~=0 & Error>Tol & Cont<Iter
```

```
    Xn = g(Xo);
```

```
    Yn = f(Xn);
```

```
    Error = abs((Xn-Xo)/Xn);
```

```
    %Error = abs(Xn-X0);
```

```
    Cont = Cont+1;
```

```
    Z(Cont,1) = Cont;
```

```
    Z(Cont,2) = Xn;
```

```
    Z(Cont,3) = Yn;
```

```
    Z(Cont,4) = Error;
```

%las z son las posiciones asignadas en la tabla a los resultados que se observaron

```
    Xo = Xn;
```

```
end
```

```
if Yn == 0
```

```
    fprintf('\n\nSOLUCION:\n')
```

```
    fprintf('%g es raiz\n\n',Xo);
```

```
else
```

```
    if Error<Tol
```

```
        fprintf('\n\nSOLUCION:\n')
```

```
        fprintf('%g es una aproximacion con un tolerancia de %g\n\n',Xo,Tol);
```

```
    else
```

```
        fprintf('\n\nSOLUCION:\n')
```

```
        fprintf('Fracaso en %g iteraciones\n\n',Iter);
    end
end
fprintf('TABLA\n\n Cont      Xn      Yn      Error Relativo\n\n')
disp(Z1);
disp(Z);
ezplot(f); %Muestra la funcion graficada
grid on %Muestra cuadrícula en la grafica de la función
```

METODO DE PUNTO FIJO TEST

METODO PUNTO FIJO

ingrese el valor inicial
0

ingrese el numero de iteraciones
100

ingrese la tolerancia que desea
0.0000001

ingrese la funcion en comillas simples
'x^2-3'

ingrese la funcion despejada en comillas simples
'abs(sqrt(3))'

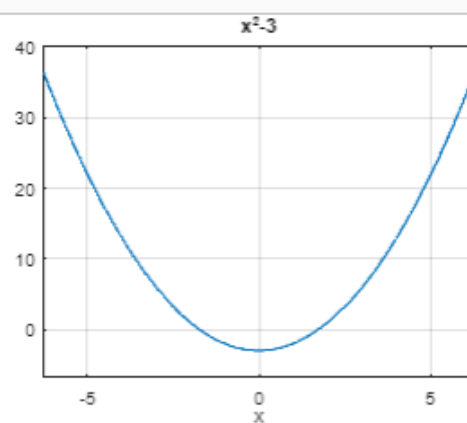
SOLUCION:

1.73205 es una aproximacion con un tolerancia de 1e-07

TABLA

Cont	Xn	Yn	Error Relativo
0	0	-3.0000000000000000	1.0000001000000000
1.0000000000000000	1.732050807568877	-0.0000000000000000	1.0000000000000000
2.0000000000000000	1.732050807568877	-0.0000000000000000	0

Figure 1 x +



Command Window

SOLUCION:

1.73205 es una aproximacion con un tolerancia de 1e-07

TABLA

Cont	Xn	Yn	Error Relativo
0	0	-3.00000000000000	1.00000010000000
1.00000000000000	1.732050807568877	-0.00000000000000	1.00000000000000
2.00000000000000	1.732050807568877	-0.00000000000000	0