**Root finder program**

Mazen medhat farid 6160

Mostafa tawfik

Ahmed sahalmy 6338

**Pseudocode:**

**1-Bisection:**

if(strcmp(app.ChoosemethodDropDown.Value,'Bisection'))

app.UITable.ColumnFormat={'char','numeric','numeric','numeric','numeric',

'numeric'};

x1=app.x\_lowerEditField.Value;

x2=app.x\_upperEditField.Value;

tol=app.ToleranceEditField.Value;

imax=app.numberofiterationsEditField.Value;

eps=abs(x2-x1)/x2;

i=1;

xr = -1;

ts = tic;

while eps>tol&&i<=imax

xr=(x2+x1)/2;

app.UITable.Data(i,:)=[i,x1,x2,xr,f(xr),eps];

if i==1

app.UITable.Data(1,6)=0;

end

if f(xr)\*f(x2)<0

x1=xr;

elseif f(xr)\*f(x1)<0

x2=xr;

else

break;

end

eps=abs(x2-x1);

i=i+1;

end

app.noofiterationsEditField.Value=i-1;

app.RootEditField.Value=xr;

app.ElapsedtimeEditField.Value = toc(ts);

**2-false position:**

elseif(strcmp(app.ChoosemethodDropDown.Value,'False-Position'))

app.UITable.ColumnFormat={'char','numeric','numeric','numeric','numeric','numeric'};

x1=app.x\_lowerEditField.Value;

x2=app.x\_upperEditField.Value;

tol=app.ToleranceEditField.Value;

imax=app.numberofiterationsEditField.Value;

if(f(x1)\*f(x2)>0)

error('error no brackets');

else

xr=x1-(x1-x2)\*f(x1)/(f(x1)-f(x2));

i=1;

app.UITable.Data(1,:)=[i,x1, x2 ,xr,f(xr), 0];

ts = tic;

while(abs(f(xr))>tol)

xold=xr;

if(f(x1)\*f(xr)<0)

x2=xr;

else

x1=xr;

end

xr=x1-(x1-x2)\*f(x1)/(f(x1)-f(x2));

AbsErr=abs(xr-xold)/xr;

app.UITable.Data(i+1,:)=[i,x1, x2 ,xr,f(xr), AbsErr];

i=i+1;

end %while

app.noofiterationsEditField.Value=i-1;

app.RootEditField.Value=xr;

app.ElapsedtimeEditField.Value = toc(ts);

end %if

**3-Fixed point:**

elseif (strcmp(app.ChoosemethodDropDown.Value,'Fixed-Point'))

app.UITable.ColumnFormat={'char','numeric','numeric'};

X0 = app.x\_lowerEditField.Value;

tol = app.ToleranceEditField.Value;

imax=app.numberofiterationsEditField.Value;

Xi = X0;

Xi1 = X0;

i = 1;

ts = tic;

for i = 1:imax

Xi = Xi1;

Xi1 = f(Xi);

eps = abs((Xi1 - Xi)/Xi1);

app.UITable.Data(i,:)=[i,Xi,eps];

if eps < tol

break;

end

end

app.UITable.Data(1,3)=0;

app.RootEditField.Value=Xi1;

app.noofiterationsEditField.Value=i;

app.ElapsedtimeEditField.Value = toc(ts);

**4-Newton Raphson:**

elseif(strcmp(app.ChoosemethodDropDown.Value,'Newton-Raphson'))

app.UITable.ColumnFormat={'char','numeric','numeric','numeric','numeric'};

x=zeros;

df=diff(str2sym(app.Fun));

app.ddxEditField.Value=char(df);

d=inline(df);

x(1)=app.x\_lowerEditField.Value;

ts = tic;

for i=1:app.numberofiterationsEditField.Value

x(i+1)=x(i)-((f(x(i))/d(x(i))));

AbsErr(i)=abs((x(i+1)-x(i))/x(i+1));

app.UITable.Data(i,:)=[i,x(i), x(i+1) ,f(x(i)),AbsErr(i)];

if AbsErr(i)<=app.ToleranceEditField.Value

app.noofiterationsEditField.Value=i;

app.RootEditField.Value=x(i+1);

app.ElapsedtimeEditField.Value = toc(ts);

break;

end

end %for

end

**5-Secant:**

elseif(strcmp(app.ChoosemethodDropDown.Value,'Secant'))

app.UITable.ColumnFormat={'numeric','numeric','numeric','numeric','numeric', 'numeric', 'numeric'};

X1 = app.x\_lowerEditField.Value;

X2 = app.x\_upperEditField.Value;

X3 = 0;

i = 0;

ts = tic;

try

for i=1:app.numberofiterationsEditField.Value

X3 = X2 - f(X2) \* ((X2 - X1)/(f(X2) - f(X1)));

AbsErr =abs((X3-X2)/X3);

app.UITable.Data(i,:)=[i,X1, X2 ,f(X1), f(X2), X3 ,AbsErr];

if AbsErr <= app.ToleranceEditField.Value

break;

end

X1 = X2;

X2 = X3;

end

app.noofiterationsEditField.Value=i;

app.RootEditField.Value=X3;

app.ElapsedtimeEditField.Value = toc(ts);

catch

f = msgbox('error method diverge');

end

**Data structures used:**

UI Table:

It was helpful in organizing the result of each method and displaying them in a tabular form.

**Analysis:**

**First case F(X)= 2\*x^5 - cos(x) - x \* exp(x)**

I= 50 xl= 1 xu=2 tol= 0.000001

G(x)= (2\*x^5 - cos(x)) /exp(x)

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Time** | **iteration** | **root** |
| **Bisection** | 1.005 | 20 | 1.152 |
| **False position** | 1.564 | 50 | 1.152 |
| **Fixed point** | Error method diverge | | |
| **Newton Raphson** | 0.5953 | 19 | 1.152 |
| **Secant** | 0.502 | 8 | 1.152 |

**second case F(X)=0.5\*exp(-x)-x**

I= 50 xl=0 xu=2 tol= 0.000001

g(x)=0.5\*exp(-x)

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Time** | **iteration** | **root** |
| **Bisection** | 0.8378 | 21 | 0.3517 |
| **False position** | 0.1606 | 4 | 0.3517 |
| **Fixed point** | 0.5494 | 15 | 0.3517 |
| **Newton Raphson** | 0.1912 | 4 | 0.3517 |
| **Secant** | 0.1673 | 5 | 0.3517 |

**third case F(X)=exp(-x)-x**

I= 50 xl=0 xu=2 tol= 0.000001

g(x)=exp(-x)

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Time** | **iteration** | **root** |
| **Bisection** | 0.6584 | 21 | 0.5671 |
| **False position** | 0.3558 | 12 | 0.5671 |
| **Fixed point** | 0.8303 | 27 | 0.5671 |
| **Newton Raphson** | 0.1501 | 4 | 0.5671 |
| **Secant** | 0.2358 | 6 | 0.5671 |

**Problematic functions:**

**Sample runs:**

**Bisection:**

**Table

Description automatically generated**

**False position:**

**Table

Description automatically generatedTable

Description automatically generated**

**Fixed point**

**Table

Description automatically generated with medium confidence**

**Newton Raphson:**

**Table

Description automatically generated**

**Secant:**

**Graphical user interface, table

Description automatically generated with medium confidence**