Practical 3: Newton-Raphson Method

Theory: x(n+1)= x(n)- f(x(n))/f'(x(n))

Q1 Perform 5 iterations of the Newton Raphson Method to find out the smallest positive root of the following equation starting with the initial approximation x_0=0.5,

f(x) = x^3-5x+1=0.

Solution:

```
(%i7) kill(all)$
      f(x):=x^3-5·x+1;
      define(df(x),diff(f(x),x));/*define (f(x_1, …, x_n), expr).
      Defines a function named f with arguments x_1, …, x_n
      and function body expr. diff (expr, x) returns the
      first derivative of expr with respect to the variable x.*/
      'x0=x0:0.5;
      n:5;
      for i:1 thru n do (
          if(equal(df(x0),0.0))
             then return()
          else
          x1:(x0-f(x0)/df(x0)),
          x0:x1,
      print("iteration",i,"root",float(x1)))$;
      print("After",n,"iterations the root is: ",float(x1))$
      wxplot2d(f(x),[x,-2,3],[y,-10,15]);
```

(%o1) $f(x):=x^3-5x+1$

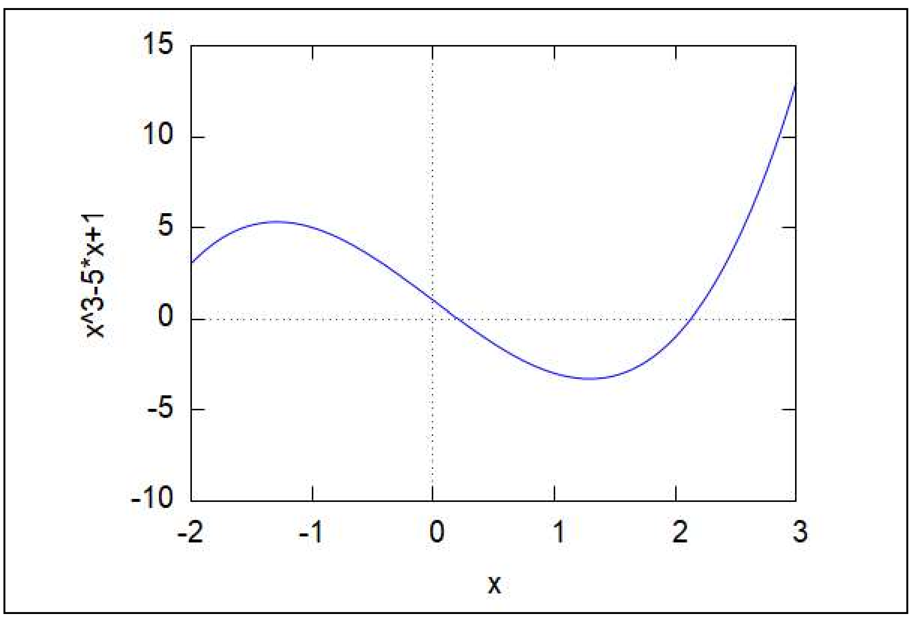
(%o2) $df(x):=3x^2-5$

(%o3) $x0=0.5$

(%o4) 5

iteration 1 root 0.1764705882352941
iteration 2 root 0.201568074338339
iteration 3 root 0.2016396750878022
iteration 4 root 0.2016396757234046
iteration 5 root 0.2016396757234047
After 5 iterations the root is: 0.2016396757234047

(%t7)



(%o7)

Q2 Perform 6 iterations of the Newton Raphson Method to find out the smallest positive root of the following equation starting with the initial approximation x_0=0.0,

f(x) = x^2-1=0.

Solution:

```
kill(all)$
f(x):=x^2-1;
define(df(x),diff(f(x),x));
'x0=x0:0.0;
for i:1 thru 6 do (
    if(equal(df(x0),0.0))
       then return()
    else
    float(x1:(x0-f(x0)/df(x0))),
    x0:x1,
print("iteration",i,"root",float(x1)))$;
print("After 6 iterations the root is: ",float(x1))$
wxplot2d(f(x),[x,-2,3],[y,-10,15]);
```

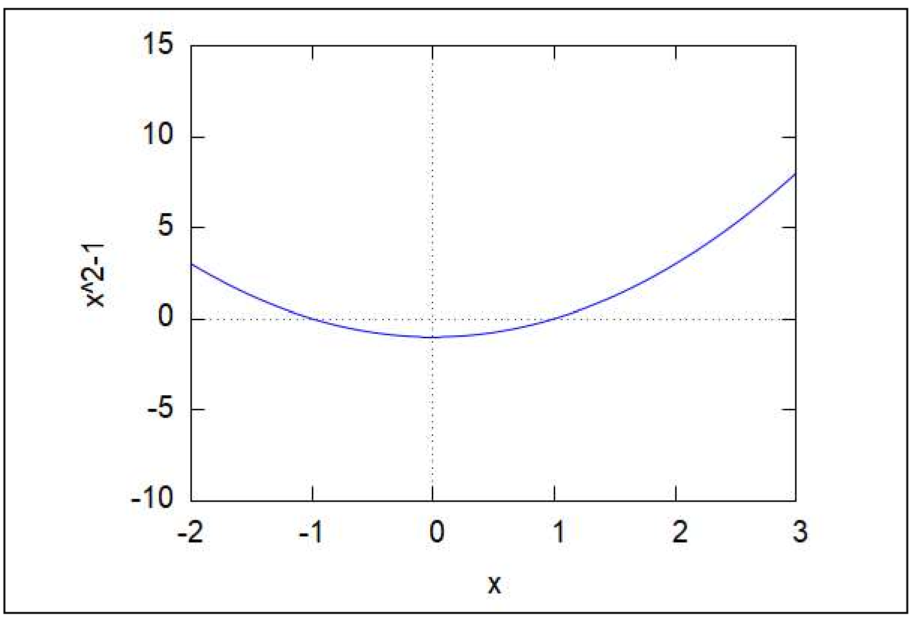
(%o1) $f(x):=x^2-1$

(%o2) $df(x):=2x$

(%o3) $x0=0.0$

After 6 iterations the root is: x1

(%t6)



(%o6)

Q3 Perform 10 iterations of the Newton Raphson Method to find out the smallest positive root of the following equation starting with the initial approximation $x_0=0.5$,

$$f(x) = x^2-1=0.$$

Solution:

```
kill(all)$
f(x):=x^2-1;
define(df(x),diff(f(x),x));
'x0=x0:0.5;
for i:1 thru 10 do (
    if(equal(df(x0),0.0))
       then return()
    else
    float(x1:(x0-f(x0)/df(x0))),
    x0:x1,if(equal(f(x0),0.0)) then return(x1) else
print("iteration",i,"root",float(x1)),
p:i);
print("After", p, "iterations the root is: ",float(x1))$;
wxplot2d(f(x),[x,-2,3],[y,-10,15]);
```

(%o1) $f(x):=x^2-1$

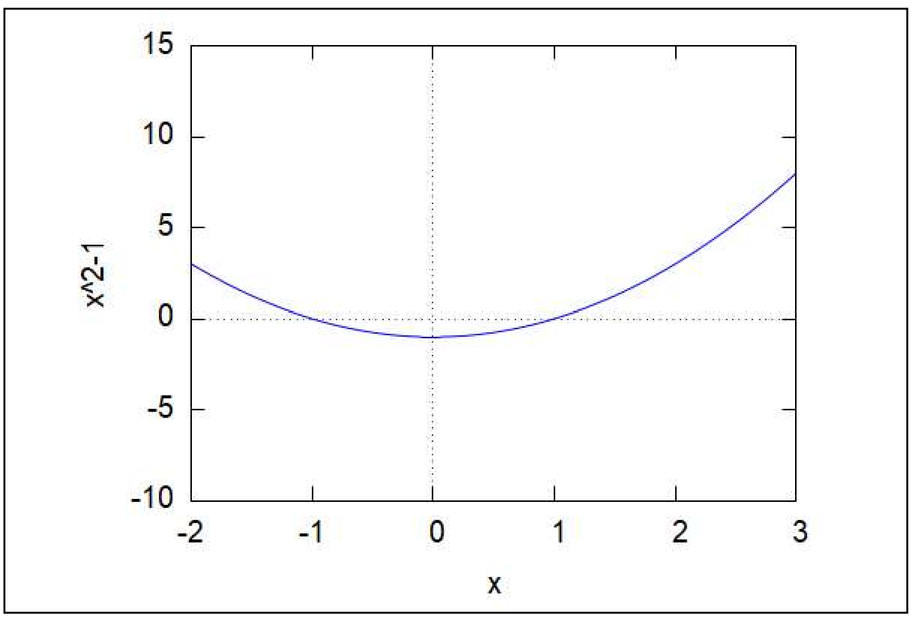
(%o2) $df(x):=2x$

(%o3) $x0=0.5$

```
iteration 1 root 1.25
iteration 2 root 1.025
iteration 3 root 1.000304878048780
iteration 4 root 1.000000046461147
iteration 5 root 1.000000000000001
```

(%o4) 1.0

After 5 iterations the root is: 1.0

(%t6) 

(%o6)

Q3 without extra condition if(equal(f(x0),0.0))

.

```
kill(all)$
f(x):=x^2-1;
define(df(x),diff(f(x),x));
'x0=x0:0.5;
for i:1 thru 10 do (
    if(equal(df(x0),0.0))
       then return()
    else
    float(x1:(x0-f(x0)/df(x0))),
    x0:x1,
print("iteration",i,"root",float(x1)),
p:i);
print("After", p, "iterations the root is: ",float(x1))$;
wxplot2d(f(x),[x,-2,3],[y,-10,15]);
```

(%o1) $f(x):=x^2-1$

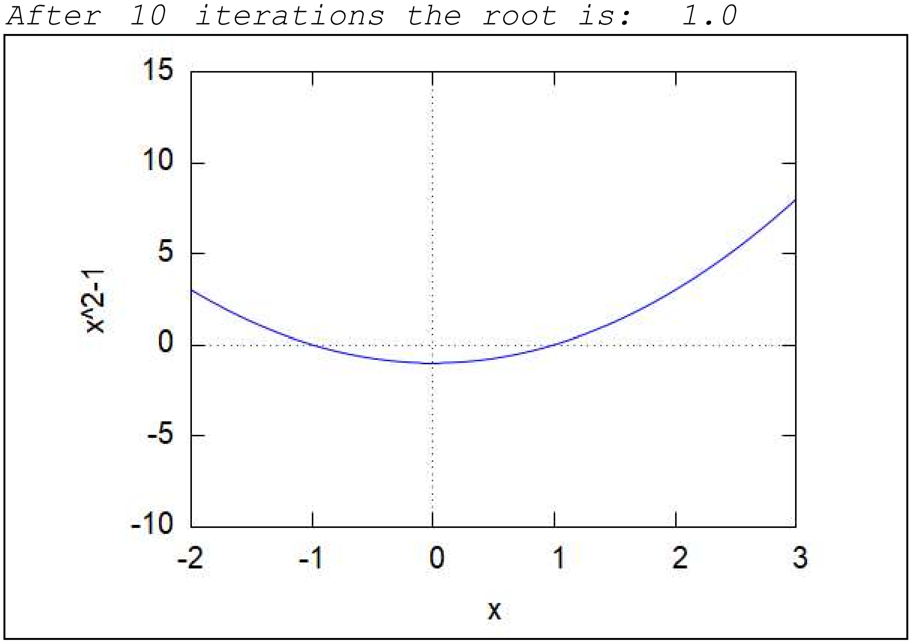
(%o2) $df(x):=2x$

(%o3) $x0=0.5$

```
iteration 1 root 1.25
iteration 2 root 1.025
iteration 3 root 1.00030487804878
iteration 4 root 1.000000046461147
iteration 5 root 1.000000000000001
iteration 6 root 1.0
iteration 7 root 1.0
iteration 8 root 1.0
iteration 9 root 1.0
iteration 10 root 1.0
```

(%o4) done

After 10 iterations the root is: 1.0

(%t6) 

(%o6)