Practical 1: Bisection method

Theory: The bisection method is used to find the approximate root of a function. It separates the interval and subdivides the interval in which the root of the equation lies. The principle behind this method is the intermediate theorem for continuous functions.

Intermediate value theorem: Let f be a continuous function over a closed interval [a, b], then the function attains every value between f(a) and f(b) i.e. for each y in between f(a) and f(b) there exists x in (a,b) such that f(x)=y. If f(a)f(b) < 0 i.e. f(a) and f(b) are of opposite signs then there exists some x in (a,b) such that f(x)=0.

Bisection method Steps:

Step 1. To apply this method, f(a)f(b) < 0 or else we can not proceed. Let f(a)f(b) < 0.

Step 2. Find the midpoint of a and b, say “t”.

Step 3. Divide the interval [a, b] as follows: If f(t)f(a) <0, there exist a root between t and a, so new interval is (a,t); else if f(t)f(b) < 0, there exist a root between t and b, so new interval is (t,b).

Step 4. Repeat the above two steps until exact root is obtained or no. of iterations are exhausted.

Q1 Perform 10 iterations of the Bisection method to obtain a real root of the following equation:
f(x)= x^3-5x+1 = 0 in the interval (0,1).

Solution:

```
(%i6) kill(all)$
      'x0=x0:0.0$
      'x1=x1:1.0$
      n:10;
      f(x):= x^3- 5 . x + 1;
      if(float((f(x0)·f(x1))>0)) then
       print("change values")
      else
      for i:1 thru n do
      (a:(x0+x1)/2,if(f(a)=0.0) then
         return(a) else /*return (a) may be used to exit explicitly
          from the current block, while, for or do loop
          bringing its argument */
       if(f(a)·f(x1))>0
          then x1:a /*interval is [x0,a]*/
       else x0:a /*interval is [a,x1]*/,
          print(i,"iteration gives ",a));
      print("The root is", a)$
```

(%o3) 10

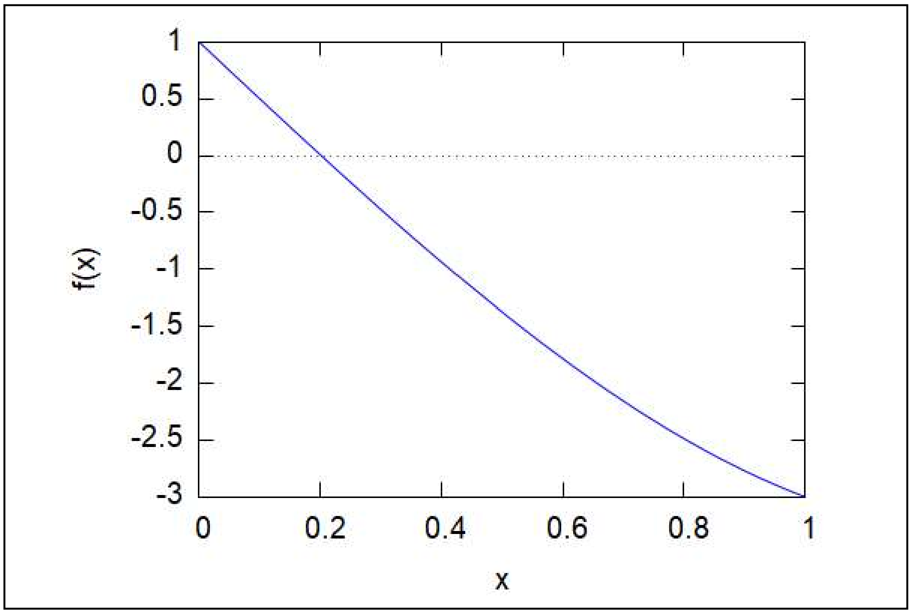
(%o4) $f(x):=x^{3}-5\ .\ x+1$

1 iteration gives 0.5
2 iteration gives 0.25
3 iteration gives 0.125
4 iteration gives 0.1875
5 iteration gives 0.21875
6 iteration gives 0.203125
7 iteration gives 0.1953125
8 iteration gives 0.19921875
9 iteration gives 0.201171875
10 iteration gives 0.2021484375

(%o5) done

The root is 0.2021484375

```
➔ wxplot2d(f(x),[x,0,1.0],[ylabel,"f(x)"]);
```

(%t7) 

(%o7)

Q2 Perform 6 iterations of the Bisection method to obtain a real root of the following equation:
f(x)= x^2-1 = 0 in the interval (0,2).

Solution:

```
(%i6) kill(all)$
'x0=x0:0.0$
'x1=x1:2.0$
n:6;
f(x):= x^2- 1;
if(float((f(x0)·f(x1))>0)) then
 print("change values")
else
for i:1 thru n do
(a:(x0+x1)/2,if(f(a)=0.0) then
   return(a) else
 if(f(a)·f(x1))>0
    then x1:a
 else x0:a,print(i,"iteration gives ",a));
print("The root is", a)$
```

(%o3) 6

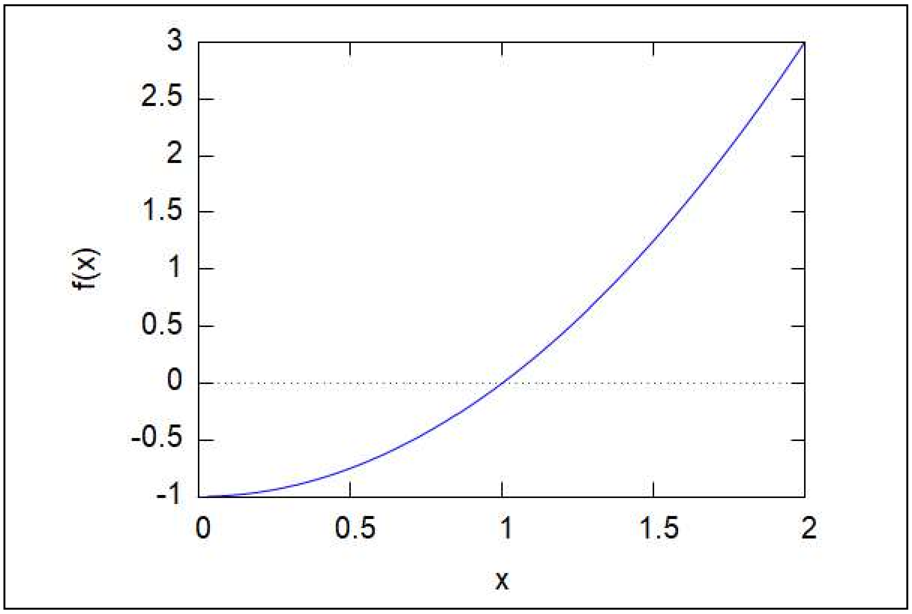
(%o4) $f(x):=x^{2}-1$

(%o5) 1.0

The root is 1.0

→

→
```
wxplot2d(f(x),[x,0,2.0],[ylabel,"f(x)"]);
```

(%t8) 

(%o8)

Assignment: Do two similar questions.