Problem Set 1

Rippy

August 2020

Problem 1

The Bisection Method and False Position Method were implemented in Mathematica.

a

Function: $f(x) = (x-1)^3$

```
Bisection[f, a, b]
```

Approx	Error
1.25	0.25
0.875	0.125
1.0625	0.0625
0.96875	0.03125
1.01563	0.015625
0.992188	0.0078125
1.00391	0.00390625
0.998047	0.00195313
1.00098	0.000976563
0.999512	0.000488281

```
FalseP[f, a, b]
```

Approx	Error
0.666667	0.333333
0.714286	0.285714
0.74359	0.25641
0.764419	0.235581
0.780365	0.219635
0.793152	0.206848
0.803739	0.196261
0.812714	0.187286
0.820463	0.179537
0.82725	0.17275

b

Function: $f(x) = x^3 + 9x - 10$

```
Bisection[f, a, b]
```

Approx	Error
1.25	0.25
0.875	0.125
1.0625	0.0625
0.96875	0.03125
1.01563	0.015625
0.992188	0.0078125
1.00391	0.00390625
0.998047	0.00195313
1.00098	0.000976563
0.999512	0.000488281

```
FalseP[f, a, b]
```

Approx	Error
0.877193	0.122807
0.969328	0.0306719
0.992332	0.00766753
0.998083	0.00191687
0.999521	0.000479219
0.99988	0.000119805
0.99997	0.0000299512
0.999993	7.48779×10^{-6}
0.999998	1.87195×10^{-6}
1.	4.67987×10^{-7}

c

For part (a), bisection converged much faster than the false position method, which had a bad error, and wasn't even close to 1. The false position method struggled with this function, since it was a bit too 'curvy' for the method to work properly. For part (b) they both converged fast. False position converged faster, because the slope near the root was far more linear, which is where the method excels.

Problem 2

Since the Bisection method's maximum error halves after every iteration, given an initial bracket of $[a, b]$, we would take the length of the bracket, that is, $\mid b - a \mid$, and since every subsequent iteration halves the maximum error of the previous, we would calculate the maximum error of the nth iteration as $\frac{1}{2^n} \mid b - a \mid$. Using this, we could have a maximum error tolerance, ϵ, and solve for how many iterations we would need to guarantee this error tolerance. We have $\epsilon = \frac{1}{2^n} \mid b - a \mid$, solving for n, we have $n = \log_2(\frac{|a-b|}{\epsilon})$

Problem 3

a

An interval on which the root (0) lies, is [-0.5,1].

b

Using the autostop code (Made in mathematica), we find the root with a maximum possible error of 10^{-5}.

Approx	Error	Max Error
0.25	0.25	0.75
−0.125	0.125	0.375
0.0625	0.0625	0.1875
−0.03125	0.03125	0.09375
0.015625	0.015625	0.046875
−0.0078125	0.0078125	0.0234375
0.00390625	0.00390625	0.0117188
−0.00195313	0.00195313	0.00585938
0.000976563	0.000976563	0.00292969
−0.000488281	0.000488281	0.00146484
0.000244141	0.000244141	0.000732422
−0.00012207	0.00012207	0.000366211
0.0000610352	0.0000610352	0.000183105
−0.0000305176	0.0000305176	0.0000915527
0.0000152588	0.0000152588	0.0000457764
-7.62939×10^{-6}	7.62939×10^{-6}	0.0000228882
3.8147×10^{-6}	3.8147×10^{-6}	0.0000114441
-1.90735×10^{-6}	1.90735×10^{-6}	5.72205×10^{-6}

Problem 4

a

We can rewrite $\mid p_n - p \mid$ as $\mid g(p_{n-1}) - g(p)$. Further, using the MVT, we can rewrite this as $\mid g'(\xi_n)(p_{n-1} - p) \mid$. We know that $g'(\xi_n) = K$. Given we know that there exists a K such that $1 < K \leq\mid g'(x) \mid$, we can substitute further, and we get the inequality $\mid p_n - p \mid\geq K \mid p_{n-1} - p \mid$. Because K is always greater than 1, we are guaranteed that $\mid p_n - p \mid>\mid p_{n-1} - p \mid$.

b

What this practically means, is given the conditions outlined, a fixed point iteration will not converge because the error only continues to increase with each subsequent iteration.

Problem 5

a

The fixed points of $g(x)$ are $x = 0, 0.5$

b

We know via a theorem we proved in class that if $\mid g'(x) \mid< 1$ around the fixed point, the iteration will converge. $g(x) = 2x(1-x)$, so $g'(x) = 2-4x$. This means it will never converge around the point 0, since the absolute value of the derivative is never less than 1 around the point. For the fixed point of 0.5, so long as you start within less than ± 0.75 of the point, you will converge. This is a reasonably close starting guess.

Problem 6

a

$g(x)$ is given by:

$$g(x) = x - \frac{f(x)}{f'(x)}$$

When we substitute $f(x), f'(x)$, we are left with:

$$g(x) = x - \frac{x^2 - a}{2x} = x - (\frac{x}{2} - \frac{a}{2x}) = \frac{x}{2} + \frac{a}{2x}$$

b

```
Newtons[f, 3.]
```

Approx	Error
2.66667	0.0209154
2.64583	0.0000820223
2.64575	1.27137×10^{-9}
2.64575	0.
2.64575	0.

Problem 7

$f(x) = \frac{1}{x} - 1$

a

```
Newtons[f, 0.5]
```

Approx	Error
0.75	0.25
0.9375	0.0625
0.996094	0.00390625
0.999985	0.0000152588
1.	2.32831×10^{-10}

b

```
Newtons[f, 0.5]
```

n	Values
1	1.
2	1.
3	1.
4	1.
5	1.

Since the values all are one, the order of convergence of this method is 2, or, quadratic.

c

```
Secant[f, 0.25, 0.5]
```

Approx	Error
0.625	0.375
0.8125	0.1875
0.929688	0.0703125
0.986816	0.0131836
0.999073	0.000926971

d

```
Secant[f, 0.25, 0.5]
```

n	Values
1	1.15106
2	0.916682
3	1.05516
4	0.967242
5	1.02059

Since the values all are approaching one, the order of convergence of this method is 1.618, or, better than linear, but not quite quadratic.