

# Assignment 2

Michael Kamensky

January 29, 2023

## 1 Running mathlib-test

Here is an example of running my tests:

```
mkamensk@vera:~/cse13s/asgn2$ ./mathlib-test -a -s
e() = 2.718281828459046, M_E = 2.718281828459045, diff = 0.000000000000000
e term = 17
pi_bbp() = 3.141592653589793, M_PI = 3.141592653589793, diff = 0.000000000000000
pi_bbp() term = 11
pi_viete() = 3.141592653589793, M_PI = 3.141592653589793, diff = -0.000000000000000
pi_viete() factors = 25
pi_euler() = 3.141592558095903, M_PI = 3.141592653589793, diff = -0.000000095493891
pi_euler() terms = 10000000
pi_madhava() = 3.141592653589800, M_PI = 3.141592653589793, diff = 0.000000000000007
pi_madhava() terms = 27
sqrt_newton() = 1.414213562373095, M_SQRT2 = 1.414213562373095, diff = -0.000000000000000
sqrt_newton() iteration = 6
```

## 2 Calculation of PI

These three methods converge really quickly to desired value

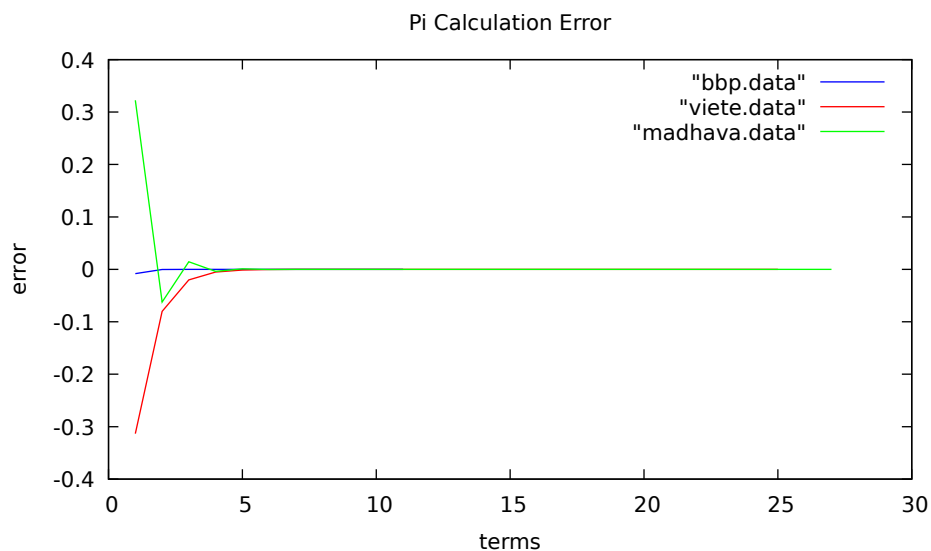


Figure 1: This is Viète, BBP and Madhava Pi Calculation Error

But Euler Pi calculation converges really slowly

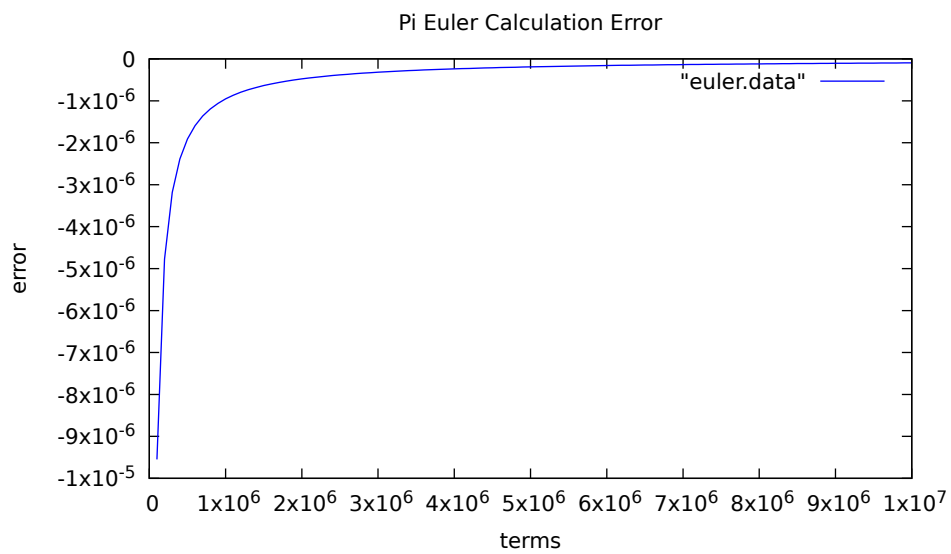


Figure 2: This is Euler Pi Calculation Error

### 3 Calculation of E

Shows the error clauclation of e()

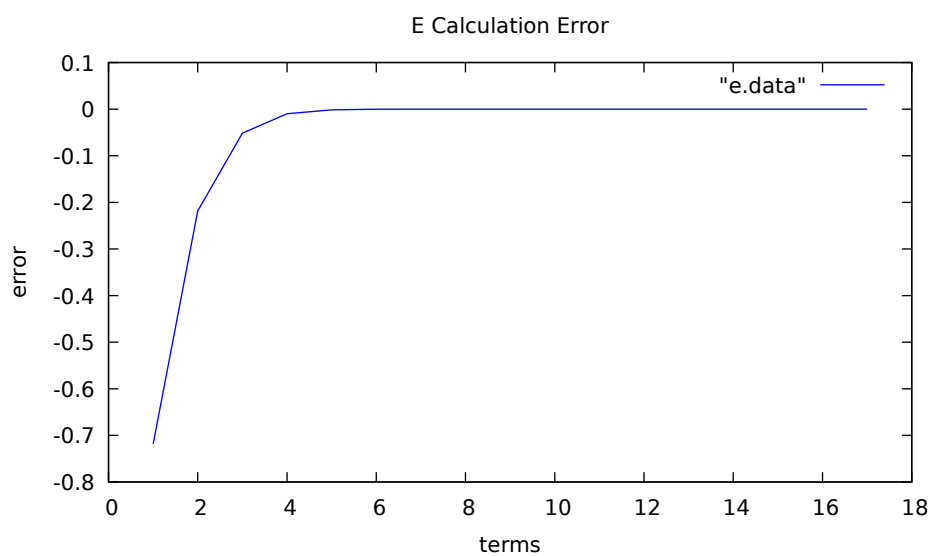


Figure 3: This is Euler E Calculation Error

### 4 Calculation of Square Root of Two

Shows the error clauclation of Sqrt 2

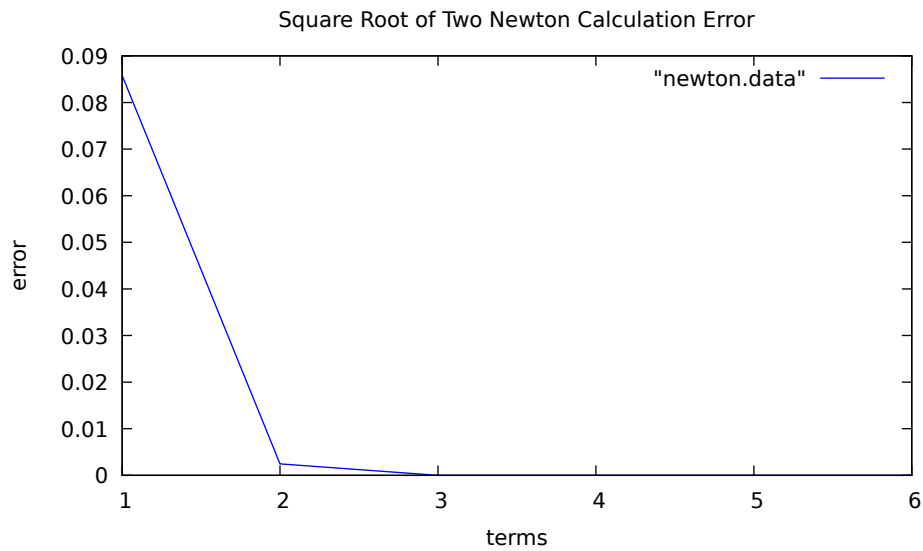


Figure 4: This is Square Root of Two Newton Calculation Error

## 5 Analysis

Calculation of pi, BBP algorithm seems the best since it converges after 11 terms. Madhava and Vietes' algorithms number of terms were pretty much the same. However, the number of terms does not mean time efficient, because of different term calculation cost. Eulers method was not only the slowest but was the least accurate. If one wants a better approximation for either pi, e, or sqrt you can lower the value of Epsilon so that the algorithms run more iterations.