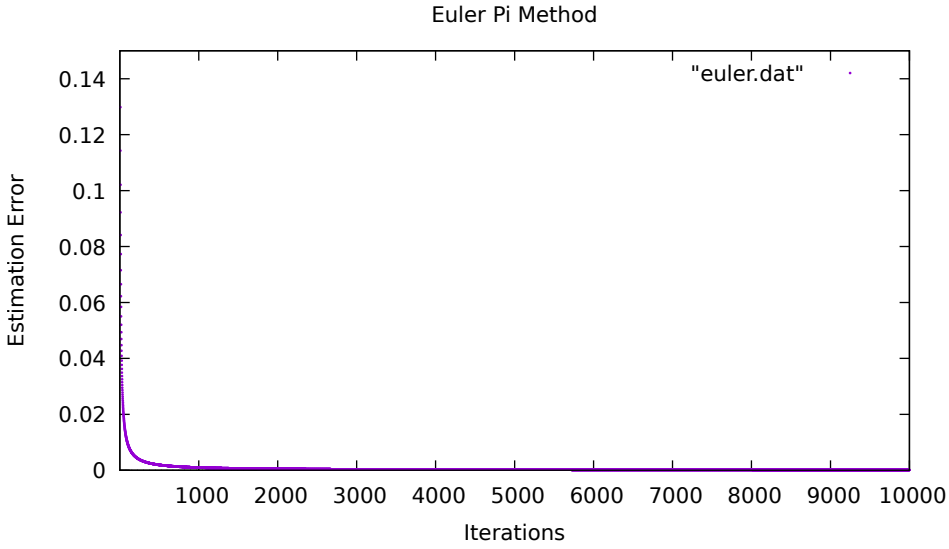
asgn2 writeup

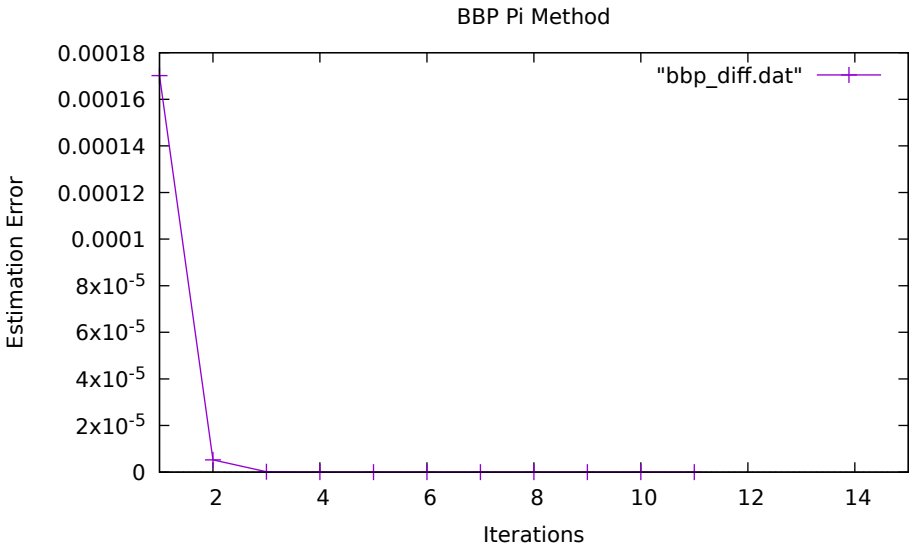
Ryan Hui

rhui1

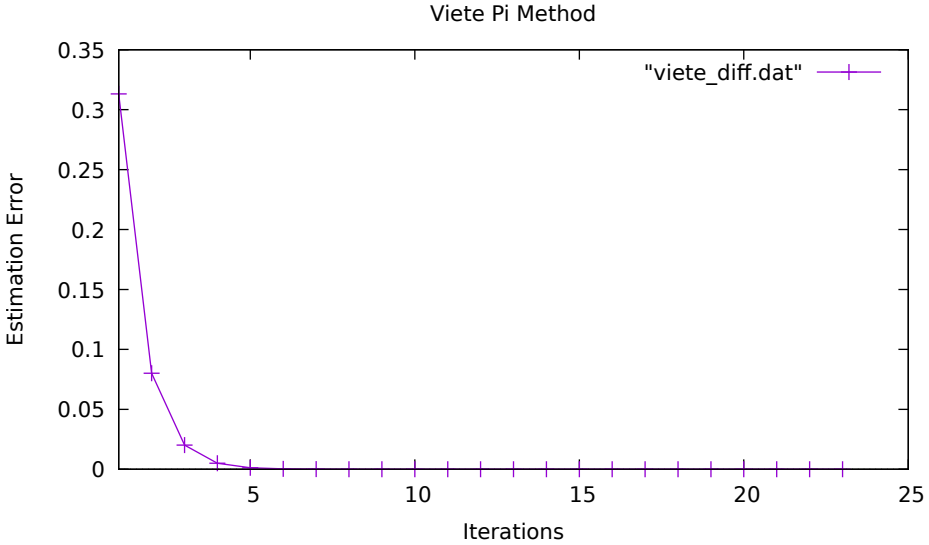
Plots



For euler pi plot, my implementation reaches 10,000,000 terms. Because of the high number, I decided to set the graph axis at 10,000 terms since the difference is so minuscule, it's not visible. Starting at term 1, the difference is around 0.07 and quickly drops down to a minuscule number. Euler is definitely the highest iterating method due to it being less effective at reaching the epsilon error.

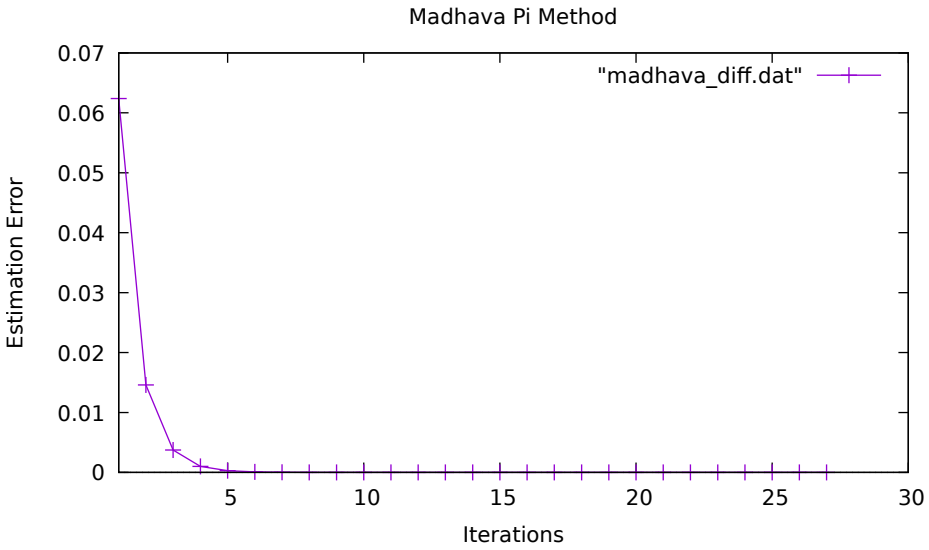


For bpp pi plot, my implementation matches the math library's pi value with a difference less than epsilon at 11 terms. I found that bbp was the lowest iteration method before reaching the pi value of the math library. Bbp also appears to be the most accurate as the difference at the first term vs pi is the smallest at around 0.00018. This shows that Bbp is the fastest and most accurate method at reaching epsilon error for pi.



For viete pi plot, my implementation matches the math library's pi value with a difference less than epsilon at 23 terms. Viete was the 2nd fastest method to find pi. Viete was accurate similar to bbp. However, Viete had the higest

starting error at around 0.32 and has a flatter rate out of all the other methods before getting near epsilon error.



For madhava pi plot, my implementation matches the math library's pi value with a difference less than epsilon at 27 terms. Madhava had similar speed to Viete and was the 3rd fastest method at finding pi. Madhava's rate was similar to Viete but it's starting error was better at around 0.062.
To close, the iteration rankings starts with bbp being the fastest and euler being the slowest. In addition, euler had the largest difference out of all my implementations and shows that it is the worse method at approximating pi.

The difference between my approximation and pi is shown by the number of iterations it took before the plot ended.