Roleen Ferrer 1731261
Professor Maxwell Dunne
CSE 13S - Computer Systems and C Programming

Assignment 2: A Small Numerical Library - Write Up

In this lab, I implemented the Sin, Cosine, and Tangent functions using the Pade approximation formulas and implemented the Exponential function using the given Taylor Series approximation, where I used an epsilon of 10^-9.

Sin

For the Sin function, I used the 14th order Pade approximation and found that the difference between my Pade approximation and the math.h library was very little. These results had a difference that was always less than 0.07, which makes sense for the given 14th order Pade approximation. If I were to have used a higher order Pade approximation, I would be able to reduce the difference and have a much close approximation in comparison to the math.h library. Since this Pade approximation centered at 0, it made sense that the difference in approximation between this and the math.h library was lower than being further away from 0. There is still some difference between each iteration, but the difference is very small and won't be seen in the program due to the difference being displayed to 10 decimal places.

Cos

For the Cos function I also used a 14th order Pade approximation. I also found that the difference between the Pade approximation and the math.h library was very little, as the difference was at least less than 0.2. Using a higher Pade approximation would have made the difference much lower, but the restriction of the 14th order Pade approximation can lead to a small difference in comparison to the math.h library. As with the Pade approximation for sin, it was also centered at 0, thus leading to more accurate results as x approaches 0. There is still a difference as it approaches 0, but the values being shown are only up to 10 decimal places.

Tan

For the Tan function, I used a Pade approximation with the set domain at [-pi/3, pi/3]. In my findings, I saw that there was 0 difference between the Pade approximation and the math.h library. This may be because the domain was restricted to this small area and the fact that the Pade approximation was to a high enough order to limit the amount of difference seen in the program. Nonetheless, there is still a little difference, though the program would not show it due to the 10 decimal place restriction.

Exp

For the Exponential function, I used a Taylor Series approximation with an epsilon of 10 ^-9. My findings saw that there was very little difference between the Taylor series approximation and the math.h library. This may be due to the fact that the epsilon that is based on 10^-9 provided a highly accurate approximation of e^x for the domain of [0,9]. Considering that, if the domain were high, say from [0,20]. These approximations would be much worse. This approximation diverges further away from e^x unless a higher epsilon was used. The higher the epsilon, the better approximation the program would get due to it's higher accuracy.

Closing Remarks

My findings of implementing Pade approximations and Taylor Series approximations functions led me to conclude that these approximations are somewhat accurate to a certain extent. The given restrictions such as the domain and epsilon led to miniscule differences in comparison to the math.h library. Ultimately, the math.h library is much more accurate since it computes to such high enough precision due to the amount of terms each function uses. Yet, it is still possible to think that the math.h library is also not exact, but very highly accurate approximations.