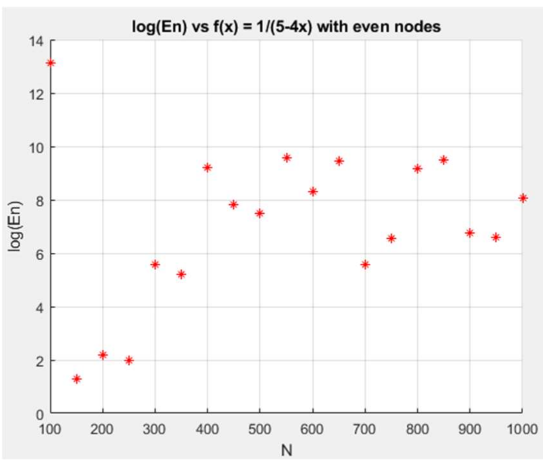
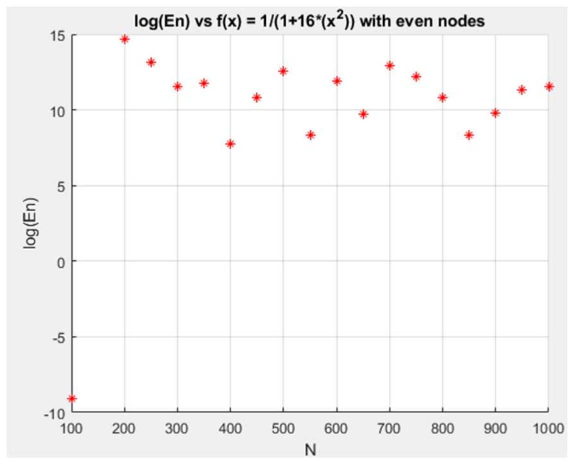
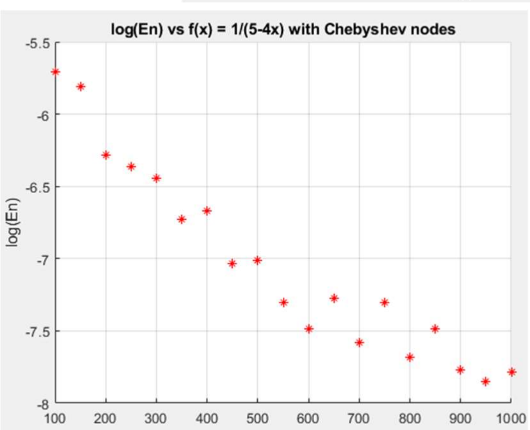
Computing Assignment 5

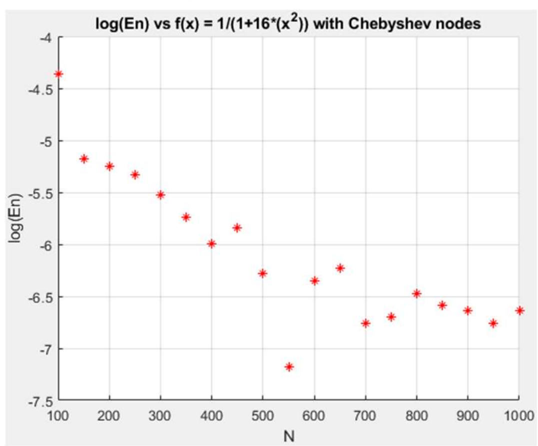
Daniel Todd
301428609
D100

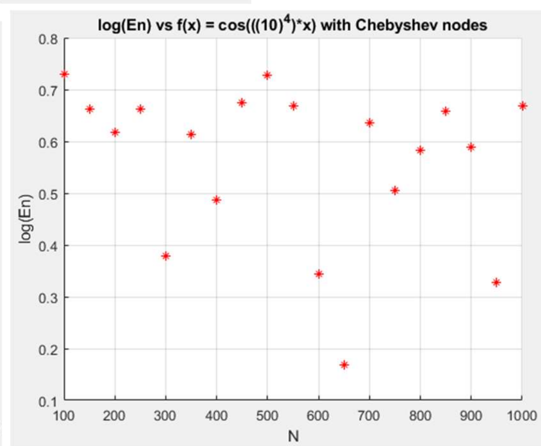
I firstly chose my n values to be $n = \{100, 150, 200, 250, \dots, 1000\}$, as these values give a large range and are evenly spaced. I then calculated equally spaced nodes on [-1, 1] given by the equation $x_i = -1 + \frac{2i}{n} \mid i = \{0, \dots, n\}$. I then calculated the interpolating polynomial for functions $f_1(x) = \frac{1}{5-4x}$ and $f_2(x) = \frac{1}{1+16\ ^2}$. I then calculated Chebyshev nodes given by $x_i = \cos\left(\frac{i\pi}{n}\right) \mid i = \{0, \dots, n\}$ for $f_1(x)$ & $f_2(x)$ as well as a third function; $f_3(x) = \cos(10^4 x)$. Then, for all 5 of my interpolating polynomials calculated, I found the error for each using the equation $e_n = \max_{-1 \le x \le 1} |P(x) - f(x)|$. I then made a plot of n vs $log_{10}(e_n)$ for each function. These plots can be seen below.











I found that the Chebyshev nodes seems to be more accurate, but do not seem to be robust against trigonometric functions. I then calculated the smallest value of n (to within ± 10) such that $e_n \le 10^{-5}$ to be $n = 155500$ for the trigonometric function f_3. Which shows the lack of robustness for Chebyshev nodes, as all other functions interpolated using Chebyshev nodes achieve this accuracy much quicker (within $n \ge 150$).