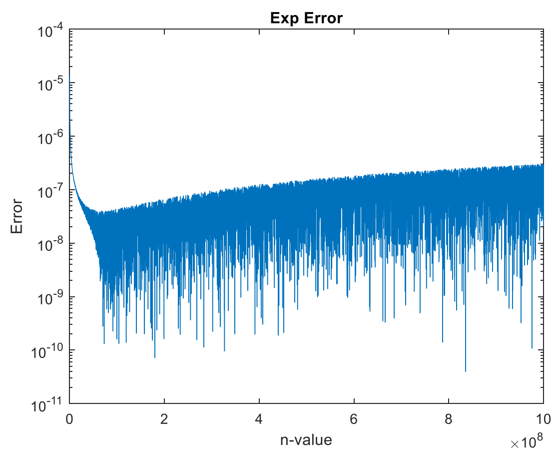
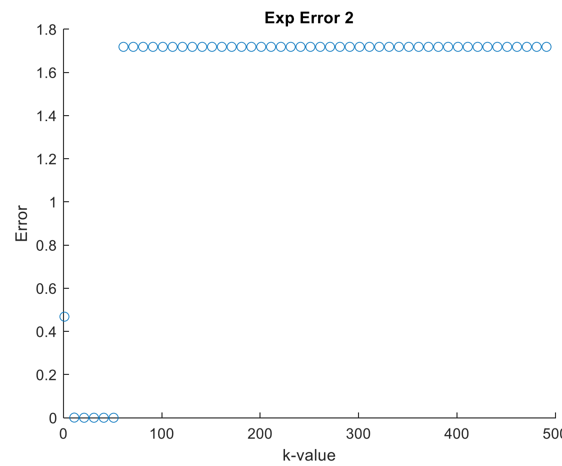
MACM 316 Assignment 1





a. In figure one (on the upper left), we can clearly see that, as the n value goes up, the absolute value of error goes up and approaches to the exponential constant of approximately 2.718. In terms of robustness, we can see that from the figure that the absolute errors are usually within a certain range and is stable and relatively small. Therefore, we can conclude that it is quite robust. In figure one, the algorithm is quite robust as the errors remain relatively small and within a certain range. In figure two (on the upper right), it is not robust as it soon renders meaningless results.

b. This phenomenon is observed that floating point algorithm and calculation could potentially render inaccuracies and rounding errors, which causes the semilogarithmic figure to fluctuate within a certain range instead of increasing steadily. When k approaches to certain value, 1/n getting too small and it approaches to 0, robustness is affected.

c. After testing with positive and negative integer, floating point number, rational and irrational numbers, I found that in terms of error, the error is within certain range as long as k-value remains small. The robustness of this algorithm remains acceptable.

Codes:

```
%For experiment 1
err=[];
n_range = 10^5:10^5:10^9;
for n=n_range
   err=[err abs(exp(1)-(1+1/n)^n)];

end



figure(1)
semilogy(n_range,err);
xlabel("n-value");
ylabel("Error");
title("Exp Error");

%For experiment 2
err=[];

k_range = 1:10:500;
e=exp(1)

for k=k_range
   n=2^k
   err=[err abs(exp(1)-(1+1/n)^n)];

end

figure(2)
scatter(k_range, err);
xlabel("k-value");
ylabel("Error");
title("Exp Error 2");
```