Mike DeLeo

Lab 1

CMPSC 455

Exercise 1:

The following exercise will determine the machine epsilon for your machine. 1/3 + 1/3 + 1/3 should equal 1, but on a machine, those numbers are represented as 0.33333… so since there is finite precision on a machine, when added up 1/3 + 1/3 + 1/3 will be appear to be 1, but the difference between 1 and this number is the machine number epsilon.

a=4/3;

b=a-1;

c=b+b+b;

eps0=1-c

Result:

eps0 = 2.2204e-16

Exercise 2:

This exercise demonstrates that the order of operations can determine how much error exists in the final answer. Specifically if a number is cancelled, then there is error propagated from the cancellation. Given

sum = 0;

for s = 1:1:7

    sum = 0;

    N = 10^s;

    for n = 1:1:N

        sum = sum + (1/n - (1/(n+1)));

    end

    sum1 = sum;

    sum = 0;

    for n = 1:1:N

        sum = sum + (1/((n+1)\*n));

    end

    x = [sum sum1 abs(1 - sum) abs(1 – sum1)];

    disp(x);

end

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| S | Sum One | Sum Two | Absolute Error 1 | Absolute Error 2 |
| 1 | 9.090909090909091e-01 | 9.090909090909089e-01 | 9.090909090909094e-02 | 9.090909090909105e-02 |
| 2 | 9.900990099009898e-01 | 9.900990099009896e-01 | 9.900990099010243e-03 | 9.900990099010354e-03 |
| 3 | 9.990009990009997e-01 | 9.990009990009996e-01 | 9.990009990002990e-04 | 9.990009990004101e-04 |
| 4 | 9.999000099990007e-01 | 9.999000099990004e-01 | 9.999000099925048e-05 | 9.999000099958355e-05 |
| 5 | 9.999900001000122e-01 | 9.999900001000117e-01 | 9.999899987844785e-06 | 9.999899988288874e-06 |
| 6 | 9.999990000010476e-01 | 9.999990000010469e-01 | 9.999989524223096e-07 | 9.999989530884434e-07 |
| 7 | 9.999998999998153e-01 | 9.999998999998143e-01 | 1.000001846884757e-07 | 1.000001856876764e-07 |

Exercise 3:

This exercise demonstrates how error can be propagated in a recursive sequence.

a = zeros(8,50);

for i = 1:1:8

    a(i,1) = 1/i;

end

b = [10 20 30 40 50];

for c = 1:1:5

    for N = 1:1:8

        for n = 2:1:b(c)

            a(N,n) = (N+1)\*a(N,n-1)-1;

        end

    end

    disp(a);

end

For this exercise there is too much data to be shown here. The matrix will be printed off for every iteration of b. The matrix is set up (N,n) so that if N= 2 and n = 5 the coordinate will be (2,5) in the matrix.

Results:

For the results, please reference the attached excel file, data.csv

What can be observed is that as the sequence grows, the values seem to grow out of control. This can be seen occurring in the set N = 30. As the sequence grows wildly large negative numbers are seen. This is a result of machine error being propagated by the sequence.

Exercise 4:

This exercise is a demonstration that changing the order of the recursion can improve accuracy. This is due to the fact that if a sequence yn = 1 + yn-1 has the next term based on the current and there is error in the current then the error will be propagated.

For the incomplete gamma given here:

The integration by parts is given by:

Where

The following recursion is based on the integration by parts of the incomplete gamma. This recursion goes from y0 to y20.

f(1)=1-1/eps;

%start from y0.

for n = 2:1:20

    %this is the constant before the integration

    %x = (-N+n)\*(1^N\*eps^-1) + (-N+n)\*(0^N\*eps^0);

    %this is the initial condition y0. This is the first ter

    f(n) = -1/eps + n\*f(n-1);

    disp(f(n));

end

clear f;

for i=1:1:5

    f(1,i) = 0;

end%initial values

%The order is reversed, so 1 is N, and the bottom is 1

b = [20 22 24 26 28 30];

for i = 1:1:5

    for n =2:1:b(i)

        %this is the constant before the integration

        f(n,i)= (f(n-1,i) + 1/eps)/n;

    end

end

disp(f);

For the first sequence, I observe that the recursion has a lot of error. This is due to the fact that the error introduced in each step is brought along to the next step. This propagation is exponentially correlated as discussed in class. En = C^n \* E0.

Refer to the data sheet attached (sheet 4a). It can be shown that the sequence begins at a number to the order of 10^15, but ends at a number to the order of 10^34. The cause of this is the propagated error introduced by basing the next number off of the last and carrying the error.

For the second sequence I noticed that the recursion did not include a lot of error. It seemed to be a lot more accurate. This supports what we learned In class where if the recursion is reversed, so is the correlation for error. So instead of the error being exponentially correlated it is the inverse.

Refer to the data sheet (sheet 4b). It is shown that the reversed recursion is much more controlled, even for iterations of size 30.