# Computing $\log(1+x)$

## The Maclaurin Series

I looked up the series expansion for the natural logarithm (http://en.wikipedia.org/wiki/Taylor_series) and found

$$\log(1-x) = -\sum_{n=1}^{\infty} \frac{x^n}{n} \qquad \text{for } -1 \leqslant x < 1. \tag{1}$$

## Naive Code

A straightforward implementation of (1) would look something like the following code.

```
function [result] = naive(x)
    N = 100;

    sum = 0;
    for n = 1:N
        sum = sum + x^n / n;
    end

    result = -sum;
end
```

To check the code I can compute the error relative to the built in log function.

```
>> naive(.5) - log(1-.5)

ans =

   2.2204e-16
```

## Smart Code

Applying Horner's Rule to (1) the code would look something like the following.

```
function [result] = smart( x )
    N = 100;

    sum = x/N;
    for n = N-1:-1:1
        sum = (sum + 1/n) * x;
    end

    result = -sum;
end
```

Again, to check the code I can compute the error relative to the built in log function.

```
>> smart(0.5) - naive(0.5)

ans =

  -2.2204e-16
```

## Time Comparison

```
>> tic; for j = 1:1000000; naive(0.5); end; toc;
Elapsed time is 26.818626 seconds.

>> tic; for j = 1:1000000; smart(0.5); end; toc;
Elapsed time is 2.168109 seconds.

>> tic; for j = 1:1000000; log(0.5); end; toc;
Elapsed time is 0.057891 seconds.
```