The Exponential Function

By Christian Haag Frederiksen

June 28, 2021

1 The exponential function

The exponential function $\exp(x) = e^x$ is often defined by the power series

$$\exp(x) = \sum_{k=0}^{\infty} \frac{x^k}{k!} \ . \qquad (1)$$

This power series has an infinite convergence radius, so the series converges for all complex numbers and thus we can use the exponential function on all numbers $z \in \mathbb{C}$

2 An implementation of the exponential function

In this exercise we consider the following implementation of the exponential function

```
#include<math.h>

double ex(double x){
    if(x<0)return 1/ex(-x);
    if(x>1./8)return pow(ex(x/2),2);
    return 1+x*(1+x/2*(1+x/3*(1+x/4
        *(1+x/5*(1+x/6*(1+x/7*(1+x/8
        *(1+x/9*(1+x/10)))))))));
}
```

In this implementation of the exponential function we create a function which take a parameter of type double and returns a value of type double. We start by checking if the parameter given is negative. If it is negative we want to return the inverse of the function value with the positive value of the parameter as input. We achieve this by calling our function recursively and finding the inverse.

The implementation approximates the exponential function by calculating the Taylor expansion around zero, thus the approximation will be most precise for small values. To get the most accuracy we recursively call our function with the original parameter divided by two and square the returned value. Due to the nature of exponentials the act of dividing the parameter and squaring the returned value cancel out eachother and thus we can use this method to calculate the value of the exponential function for some parameter by calculating the exponential function for a much smaller value. In the implementation we do this trick until we have transformed the original parameter into a number smaller than $\frac{1}{8}$, which determines the accuracy of the implementation.

Finally, if we call the function with a parameter which is positive and smaller than $\frac{1}{8}$, we calculate the Taylor series.

Instead of using exponents, we simply multiply the value onto it self and arbitrary number of times. The time complexity of this operation is low, so it is very fast. It has the added advantage that we are not computing factorials, that is, very large numbers, that will conflict with our precision very quikly. Hence the computer will not be forced to add up numbers of different orders of magnitude, and we retain precision.

3 Figures

Here is an illustration of the numerical implementation that was described above in figure (1) shows the ”pdf” terminal of Pyxplot.

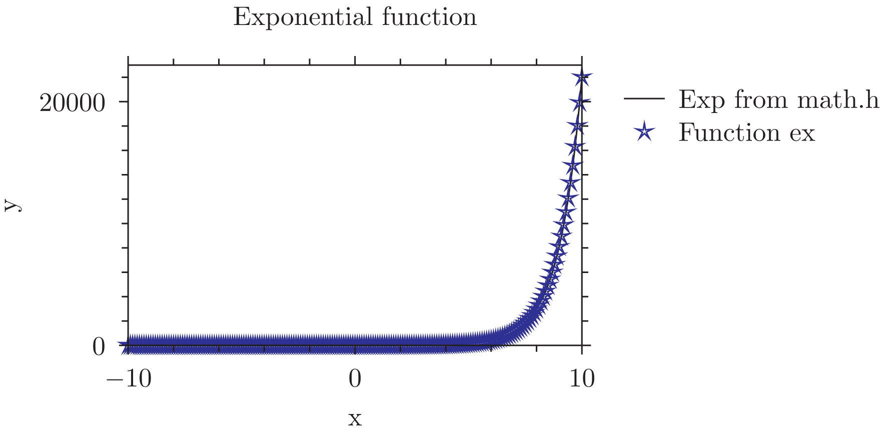

Figure 1: Plot of numerical implementation as described above, versus the standard implementation found in the header math.h. Notice the two graphs are indistinguishable to the naked eye.