

Challenge: Exponential function approximation

Exponential functions are of great importance in mathematics, physics and engineering.

One of the most important functions is e^x . It has the very important property that it is the only function that its derivative is the same as itself.

$$e^x = \frac{d(e^x)}{dx}$$

So any process where the rate of change of a quantity is proportional to the quantity itself will be described by exponential functions. Finance, population growth, oscillators all have these characteristics.

When we want to model these processes in a computer we have a problem. e is irrational and that means that we have to calculate approximate values. Specially our digital computers only know how to solve simple operations... so how can we calculate the value of complex equations?

The answer is to simplify the functions into operations a computer can manage without any problem. sums, multiplications and divisions are simple enough.

Let's use the following formula to estimate the value of e up to 9 decimals only using multiplications and sums.

$$e^x = \sum_{n=0}^{\infty} \frac{x^n}{n!} = 1 + \frac{x}{1} + \frac{x^2}{1 * 2} + \frac{x^3}{1 * 2 * 3} + \dots$$

As you can see x is only raised to integer powers (i.e. multiplications) and $n!$ stands for factorial, which is the integer multiplication up to n (note that $0!$ and $1!$ is equal to 1).

1. Build a program that given x and n would approximate e^x using the identity above. That is only using sums, multiplications and divisions (exponentials are not allowed by definition). You can use any programming language you like (it could be excel if you want).
2. What should be the value of n in order to approximate e^1 up to 9 decimals?

Please send your answers through email or a private repository to the github account of the mentor. Be mindful of sending the code well commented so that it will be easy to understand and test the program.