Contents

Recursion with Peano

Let P be the predecessor function and S the successor. Given the following implementation in Haskell of the natural numbers:

```
data Natural = Zero | S Natural
```

Each of the following examples are implemented in math and Haskell syntax, using a **recursive** approach.

Addition

```
add :: Natural -> Natural -> Natural
add Zero n = n
add (S m) n = add m (S n)
```

$$\text{add}(a,b) = \begin{cases} a & ,b=0 \\ \text{add}(a+1,b-1) & \end{cases}$$

Multiplication

```
mul :: Natural -> Natural -> Natural
mul Zero n = Zero
mul (S m) n = add n (mul m n)
```

$$\text{mul}(m,n) = \begin{cases} 0 & ,m=0 \\ \text{sum}(n,\text{mul}(m-1,n)) & \end{cases}$$

Subtraction

```
add :: Natural -> Natural -> Natural
add Zero n = n
add (S m) n = add m (S n)
```

$$\text{sub}(m,n) = \begin{cases} m & ,n=0 \\ ! & ,m=0 \\ \text{sub}(m-1,n-1) & \end{cases}$$