Factorial, fibonacci, first and second procedures

Eduardo Acuña Yeomans

This file contains four procedure definitions: factorial and fibonacci are well known clasic recursive functions but first and second aren't usefull at all, they just select one of the two arguments.
Part of the test for the cactus program.

factorial : *number* $\rightarrow$ *number*
Return the factorial of the number n, where $n! = n * (n-1) * ... * 1$.

```
(define (factorial n)
  (if (= n 0)
      1
      (* n (factorial (- n 1)))))
```

fibonacci : *number* $\rightarrow$ *number*
Return the n^{th} number of the fibonacci sequence.

```
(define (fibonacci n)
  (if (or (= n 0)
          (= n 1))
      1
      (+ (fibonacci (- n 1)) (fibonacci (- n 2)))))
```

first : *number* $\times$ *number* $\rightarrow$ *number*
Return the first argument.

```
(define (first a b) a)
```

second : *number* $\times$ *number* $\rightarrow$ *number*
Return the second argument.

```
(define (second a b) b)
```