



Fibonacci - Comparativo de Tempo de Execução e Memória

ESTCMP005 - PLP - Atividade 007 por Gabriel Sena San Martin

Fibonacci Recursivo Sem Cauda

```
module Fibonacci where
```

```
fibonacci :: (Eq a, Num a, Num p) => a -> p
```

```
fibonacci 0 = 0
```

```
fibonacci 1 = 1
```

```
fibonacci n = fibonacci (n - 1) + fibonacci (n - 2)
```

Fibonacci Recursivo Com Cauda

```
{-# LANGUAGE BangPatterns #-}
```

```
fibonaccitail :: (Eq t, Num t, Num b) => t -> b
```

```
fibonaccitail n = go n (0,1)
```

```
    where
```

```
        go !n (!a, !b) | n == 0 = a
```

```
                    otherwise = go (n-1) (b,a+b)
```



Fibonacci

-	Recursivo sem cauda	Recursivo com cauda
10	(0,03 segundos, 121,47 KBs)	(0,01 segundos, 64,05 KBs)
20	(0,05 segundos, 7,72 MBs)	(0,01 segundos, 69,60 KBs)
30	(1.49 segundos, 943,12 MBs)	(0,01 segundos, 75,16 KBs)
100	' - '	(0,01 segundos, 115,94 KBs)
1000	; - ;	(0,01 segundos, 700,18 KBs)