

# Combinatore $Y$

Funzione di ordine superiore che permette di implementare la ricorsione nei linguaggi che non la supportano, come il  $\lambda$ -calcolo:

$$Y = \lambda f.(\lambda x.f(xx))(\lambda x.f(xx))$$

Ha come proprietà fondamentale:

$$YF \equiv_{\beta} F(YF)$$

## Dimostrazione

$$\begin{aligned} YF &= (\lambda f.(\lambda x.f(xx))(\lambda x.f(xx)))F \\ &\rightarrow (\lambda x.F(xx))(\lambda x.F(xx)) \\ &\rightarrow \underline{F((\lambda x.F(xx))(\lambda x.F(xx)))} \\ &\leftarrow F(\lambda f.(\lambda x.f(xx))(\lambda x.f(xx)))F \\ &= F(YF) \end{aligned}$$

Notare che comunque  $YF \not\rightarrow^* F(YF)$ , l'ultima freccia è nella direzione opposta.

## Utilizzo

$$\begin{aligned} F &= \lambda f.\lambda n. \text{ If } (\text{IsZero } n) \ C_1 \ (\text{Times } n \ (f \ (\text{Pred } n))) \\ \text{Factorial} &= YF \end{aligned}$$

$F$  riceve in input una funzione  $f$  che calcola il fattoriale fino a  $n$ , e restituisce una funzione che lo calcola fino a  $n + 1$ . Il limite dell'applicazione ripetuta di  $F$  a sé stessa restituisce una funzione che calcola il fattoriale per tutti gli  $n$  – il suo *punto fisso*, visto che applicare nuovamente  $F$  non modifica la funzione che otteniamo.