

Programmazione Funzionale

Esercitazione 3 – Ricorsione sugli Interi

Esercizio 1. Dato una funzione $f : \mathbb{N} \times \mathbb{N} \rightarrow \mathbb{N}$ definiamo $iter(f, e)$ l'iterazione di f con elemento neutro $e \in \mathbb{N}$, per induzione:

$$iter(f, e)(0) = e \quad \text{and} \quad iter(f, e)(n) = f(n, iter(f, e)(n-1)).$$

1. Scrivere in OCAML una funzione ricorsiva $iter$ che prende una funzione f e un intero e e restituisce la funzione $iter(f, e)$.
2. Scrivere in OCAML una versione ricorsiva di coda della funzione precedente.
3. Quali parametri dobbiamo dare a $iter$ per ottenere la funzione $sum : n \mapsto \sum_{0 \leq i \leq n} i$.
4. Quali parametri dobbiamo dare a $iter$ per ottenere la funzione $fact : n \mapsto \prod_{1 \leq i \leq n} i$.

Esercizio 2. La successione di fibonacci e la funzione $u : \mathbb{N}^* \rightarrow \mathbb{N}$ definita per induzione tale che;

$$u_1 = 1 \quad u_2 = 1 \quad u_{n+2} = u_{n+1} + u_n.$$

1. Definire la funzione fibonacci che calcola per ogni intero positivo n l'intero u_n .
2. Definire la versione tail recursive di fibonacci.

Esercizio 3. Considerate le funzione definite per mutua ricorsione

$$f(0) = 1 \quad f(n) = 1 + g(n-1) \quad g(0) = 0 \quad g(n) = 0 + f(n-1).$$

1. Definire le funzioni f e g in OCAML.
2. Cosa calcolano f e g ?
3. Definire queste funzioni in modo tail recursive.

Esercizio 4. Dare una versione definita per mutua ricorsione delle funzione `odd` e `even`. `odd` n ritorna `true` quando n è dispari e `false` quando n è pari, invece, `even` n ritorna `true` quando n è pari e `false` quando n è dispari.

Esercizio 5. Usando la mutua ricorsione definire la funzione che rappresenta la sequenza $(u_n)_{n \in \mathbb{N}}$ definita per induzione:

$$u_0 = 1 \quad u_{2n} = u_{2n-1} + 3 \quad u_{2n+1} = u_{2n} * 2.$$