

5 Ricorsione

Definizione 5.1 (Ricorsione). *A tempo di **compilazione**: una funzione usa il suo nome (chiama se stessa) nel suo corpo. A tempo di **esecuzione**: chiamate annidate della **stessa** funzione*

Una funzione ricorsiva è chiamata per risolvere un problema scomposto in:

- **Caso base**: la funzione restituisce un valore
- **Passo ricorsivo**: la funzione viene chiamata su un problema analogo a quello iniziale ma di dimensioni minori, avvicinandosi al *caso base*

Quando si arriva al caso base viene effettuata una sequenza inversa di return statement, combinando i risultati parziali in quello finale.

Esempio 5.1 (Fattoriale). Il fattoriale di un intero non negativo n è il prodotto degli interi positivi $\leq n$ escluso lo 0. Si indica con $n!$ e si impone per definizione $0! = 1$.

$$n! = \prod_{i=1}^n i = n * (n - 1) * \dots * 1 \quad (2)$$

oppure definita in maniera ricorsiva:

$$n! = \begin{cases} 1, & n = 0 \\ n * (n - 1)!, & n > 0 \end{cases} \quad (3)$$

In maniera programmatica possiamo scriverlo come:

```
1  func F(var n: Int) -> Int {
2      if (n-1) {
3          return 1
4      } else {
5          return n * F(n-1)
6      }
7  }
```

Listing 7: Fattoriale con ricorsione

5.1 Ricorsione e iterazione

	Ricorsione	Iterazione
Controllo di terminazione	Condizione di ricorsione	Condizione di controllo nel loop
Ripetizioni	Chiamate ricorsive della funzione	Esecuzione ripetuta del corpo dell'iterazione
Convergenza alla terminazione	I passi ricorsivi riducono il problema al caso base	Il contatore si avvicina al valore di termine
Ripetizione infinita	Il passo ricorsivo non riduce il problema e non si avvicina al caso base	La condizione di controllo non è mai falsa

Nella *ricorsione*, al contrario dell'*iterazione*, ogni chiamata alla funzione genera un nuovo record di attivazione contenente una nuova copia delle variabili e consumando lo stack di esecuzione. Questo può generare **overhead**.

In generale ogni problema *ricorsivo* può essere anche scritto *iterativamente*. È consigliato scriverlo ricorsivamente quando ciò facilita la lettura del problema stesso.