

# Passaggio argomenti

Nel `SC$++` 2003, gli argomenti delle funzioni possono essere passati secondo due modalità:

- **Passaggio per valore** (`T`): si effettua una copia del valore dell'argomento nel parametro della funzione.
- **Passaggio per riferimento a lvalue** (`const T&` oppure `T&`): il parametro della funzione è un riferimento che viene inizializzato con l'argomento stesso, senza effettuare una copia.

La linea guida, semplice, prevede di passare per copia solo gli oggetti piccoli, per i quali la copia stessa non è costosa, mentre gli oggetti potenzialmente grandi sono passati per riferimento a costante. Nel caso sia necessario modificare direttamente l'argomento in input, si opta per il passaggio a riferimento modificabile.

Per il **valore di ritorno**, generalmente si opta per il ritorno per valore, perché non si possono restituire riferimenti a variabili che sono allocate automaticamente dalla funzione (il chiamante otterrebbe dei riferimenti *dangling*). Un caso in cui si possono restituire riferimenti (e quindi evitare copie costose) è quello in cui siamo sicuri che il riferimento è ad un oggetto il cui tempo di vita continuerà sicuramente anche dopo la chiamata di funzione.

---

## lvalue vs rvalue

In `SC$++`, un **lvalue** è un'espressione che rappresenta un oggetto identificabile nella memoria del computer, cioè un valore che ha un indirizzo di memoria. Ad esempio, una variabile, un elemento di un array o una funzione che restituisce un riferimento a un oggetto sono tutti esempi di *lvalue*.

D'altra parte, un **rvalue** è un'espressione che rappresenta un valore che non ha un indirizzo di memoria associato. Ad esempio, un valore costante come `5`, una stringa letterale come `hello` o il risultato di una chiamata a una funzione che restituisce un valore sono tutti esempi di *rvalue*.

In breve, **la differenza principale tra lvalue e rvalue sta nell'indirizzo di memoria**. Gli *lvalue* sono identificabili dalla memoria e possono essere modificati, mentre gli *rvalue* sono solo valori temporanei che non possono essere modificati. Inoltre, alcune operazioni in `SC$++` richiedono un *lvalue*, come l'assegnazione di un valore a una variabile, mentre altre richiedono un *rvalue*, come l'uso di un valore costante in un'operazione aritmetica.

Approfondimento: [\[\[11-lvalue\\_rvalue\]\]](#)

**NOTA:** a partire dal `SC$++` 2011 sono stati introdotti i riferimenti a *rvalue* e, di conseguenza, la possibilità di passare un argomento per riferimento a *rvalue*. L'argomento (un po' tecnico) verrà affrontato quando parleremo della gestione delle risorse.

**NOTA:** a volte, in maniera impropria, si parla di passaggio di un argomento *"per puntatore"*. Tecnicamente, si tratta di un caso specifico del passaggio per valore (il valore del puntatore, ovvero un indirizzo). Questo modo di procedere è un'eredità del linguaggio `SC$` ed è spesso rimpiazzabile dal passaggio per riferimento. Un argomento di tipo puntatore continua ad avere senso quando l'argomento è opzionale: in questo caso, passando il puntatore nullo si segnala alla funzione che quell'argomento non è di interesse per una determinata chiamata.

[Torna all'indice](#)