

# Complessità temporale (notazione Big-O) per le principali operazioni in una lista doppiamente concatenata

📅 Created	@January 21, 2025 3:33 PM
📖 Class	Prog2

## 1. Accesso a un elemento

- **Best case:**  $O(1)$  (se si accede al primo elemento o si ha già il riferimento diretto).
- **Worst case:**  $O(n)$  (deve attraversare tutti gli  $n$  elementi).
- **Medium case:**  $O(n)$  (in media attraversa  $n/2$  elementi, quindi è comunque  $O(n)$ ).

## 2. Inserimento di un elemento

- **All'inizio della lista:**
  - **Best case:**  $O(1)$
  - **Worst case:**  $O(1)$
  - **Medium case:**  $O(1)$
- **Alla fine della lista:**
  - **Best case:**  $O(1)$  (grazie al puntatore diretto alla coda).
  - **Worst case:**  $O(1)$  (sempre costante, a differenza della lista semplice).
  - **Medium case:**  $O(1)$
- **In una posizione arbitraria:**

- **Best case:**  $O(1)$  (se l'iteratore si trova già nella posizione desiderata).
- **Worst case:**  $O(n)$  (deve attraversare la lista fino alla posizione).
- **Medium case:**  $O(n)$

### 3. Cancellazione di un elemento

- **All'inizio della lista:**
  - **Best case:**  $O(1)$
  - **Worst case:**  $O(1)$
  - **Medium case:**  $O(1)$
- **Alla fine della lista:**
  - **Best case:**  $O(1)$  (grazie al puntatore diretto alla coda).
  - **Worst case:**  $O(1)$  (sempre costante, a differenza della lista semplice).
  - **Medium case:**  $O(1)$
- **In una posizione arbitraria:**
  - **Best case:**  $O(1)$  (se il nodo da eliminare è già noto).
  - **Worst case:**  $O(n)$  (deve attraversare la lista per trovare il nodo).
  - **Medium case:**  $O(n)$

### 4. Ricerca di un elemento

- **Best case:**  $O(1)$  (se il nodo cercato è il primo o l'ultimo, a seconda del verso di ricerca).
- **Worst case:**  $O(n)$  (deve attraversare tutti i nodi).
- **Medium case:**  $O(n)$  (in media attraversa  $n/2$  elementi, ma è comunque  $O(n)$ ).

### Riassunto Tabellare

Operazione	Best Case	Worst Case	Medium Case
Accesso	$O(1)$	$O(n)$	$O(n)$
Inserimento (inizio)	$O(1)$	$O(1)$	$O(1)$

Inserimento (fine)	$O(1)$	$O(1)$	$O(1)$
Inserimento (arbitrario)	$O(1)$	$O(n)$	$O(n)$
Cancellazione (inizio)	$O(1)$	$O(1)$	$O(1)$
Cancellazione (fine)	$O(1)$	$O(1)$	$O(1)$
Cancellazione (arbitraria)	$O(1)$	$O(n)$	$O(n)$
Ricerca	$O(1)$	$O(n)$	$O(n)$

## Differenze principali rispetto a una lista concatenata semplice:

1. **Inserimento e cancellazione alla fine:** sono sempre  $O(1)$  grazie al puntatore diretto alla coda, a differenza della lista semplice, dove può essere  $O(n)$  in alcuni casi.
2. **Navigazione bidirezionale:** rende più efficienti alcune operazioni (es. ricerca o modifica partendo dalla fine).