

# Lezione 5

## Appunti di Davide Scarlata 2024/2025

 **Docente:** Michele Garetto

 **Email:** [michele.garetto@unito.it](mailto:michele.garetto@unito.it)

 **Corso:** Linguaggio C

 **Risorsa utile:** [Moodle Unito](#)

 **Data:** 12 marzo 2025

---

### Le Liste Linkate

Una lista è un **elenco** che può essere:

1. Vuoto.
2. Composto da un elemento seguito da un'altra lista (*definizione ricorsiva*).

#### ◆ Ruoli degli Elementi

Ogni elemento della lista ha un ruolo specifico:

- **Primo elemento:** *head* (testa).
- **Successore:** Nodo immediatamente successivo.
- **Predecessore:** Nodo immediatamente precedente.
- **Ultimo elemento:** *tail* (coda).

#### ◆ Tipologie di Liste

- **Ordinate:** Gli elementi seguono un ordine prestabilito.
  - **Non ordinate:** Gli elementi sono disposti in ordine casuale.
  - **Con ripetizioni:** Gli elementi possono comparire più volte.
- 

### Operazioni sulle Liste Linkate

Ecco alcune operazioni comuni che si possono effettuare sulle liste:

## ◆ Inserimento e Rimozione

Gli elementi possono essere **aggiunti o rimossi** in qualsiasi posizione:

- All'inizio (testa).
- Alla fine (coda).
- In una posizione intermedia.

## ◆ Gestione di Liste Ordinate

Se la lista è ordinata, è necessario **preservarne l'ordine** durante le modifiche.

## ◆ Percorrenza della Lista

Analizzare la lista per calcolare dati utili, come:

- **Conteggio:** Quanti elementi contiene?
  - **Calcolo:** Somma dei valori negli elementi in posizione dispari.
- 

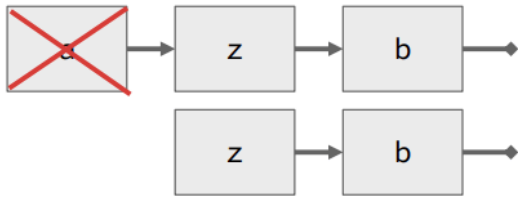
## ✗ Cancellazione di Elementi

Una **lista concatenata** è una sequenza di nodi in cui ogni nodo contiene un riferimento al successivo.

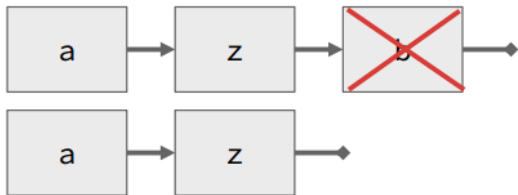
L'accesso alla lista avviene tramite una variabile puntatore.

Se una lista è vuota, il riferimento sarà `NULL`.

# LA CANCELLAZIONE MODIFICA LA LISTA



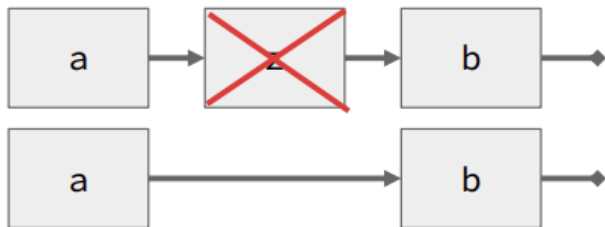
z sarà il nuovo primo elemento della lista



z sarà il nuovo ultimo elemento della lista



la lista diventa vuota



occorre collegare i pezzi rimanenti

## Struttura di un Nodo

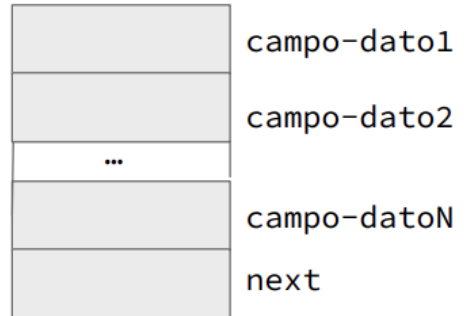
Un nodo di una lista concatenata contiene:

1. **Dato:** Un valore, ad esempio un numero o una stringa.
2. **Riferimento:** Un puntatore al nodo successivo (oppure `NULL` se è l'ultimo elemento).

# NODI DI UNA LISTA, IDEA #1

```
struct nodo {  
    tipo-dato1  campo-dato1;  
    tipo-dato2  campo-dato2;  
    ...  
    tipo-datoN  campo-datoN;  
    struct nodo* next;  
};  
struct nodo vrb;
```

## IN MEMORIA



## ◆ Esempio di Struttura in C

```
struct nodo {  
    tipo_dato campo_dato;  
    struct nodo* next;  
};  
  
struct nodo *vrb;  
vrb = (struct nodo*) malloc(sizeof(struct nodo));  
vrb->campo_dato = valore_iniziale;  
vrb->next = NULL; // Inizializzazione
```

💡 **Nota:** Per semplificare l'uso dei puntatori, si può utilizzare `typedef`.

## 🗑 Rimozione di un Nodo in Testa

Quando si rimuove il primo nodo:

- Il nodo successivo diventa la nuova testa.
- Se il successore è `NULL`, la lista diventa vuota.

## ◆ Esempio di Funzione in C

```

struct Nodo {
    int valore;
    struct Nodo* prossimo;
};

void cancellaNodoTesta(struct Nodo** testa) {
    if (*testa == NULL) {
        printf("La lista è già vuota.\n");
        return;
    }
    struct Nodo* temp = *testa;
    *testa = (*testa)->prossimo;
    free(temp);
    printf("Nodo in testa cancellato.\n");
}

```

## + Aggiunta di un Nodo in Testa

Per aggiungere un nuovo nodo all'inizio:

- Il nodo attuale diventa il successore del nuovo nodo.

### ◆ Esempio di Funzione in C

```

typedef struct Nodo {
    int valore;
    struct Nodo *prossimo;
} Nodo;

void aggiungiInTesta(Nodo **testa, int valore) {
    Nodo* nuovoNodo = (Nodo*)malloc(sizeof(Nodo));
    nuovoNodo->valore = valore;
    nuovoNodo->prossimo = *testa;
    *testa = nuovoNodo;
}

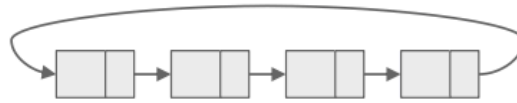
```

## Tipologie Avanzate di Liste

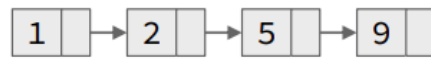
1. **Liste Circolari:** L'ultimo nodo punta al primo, formando un ciclo.

2. **Liste Ordinate:** Gli elementi sono organizzati in base a criteri predefiniti.
3. **Liste Double Ended:** Accesso sia alla testa che alla coda.
4. **Liste Doppiaemente Linkate:** Ogni nodo ha un riferimento sia al successivo che al precedente.

lista circolare:



lista ordinata:



lista double ended:



si può inserire/estrarre ad ambo le estremità

lista doppiamente linkata:

