

## Parametri di tipo vettore

- ▶ Il meccanismo del passaggio **per valore** di un **indirizzo** consente il passaggio di vettori come parametri di funzioni/procedure.
- ▶ Quando si passa un vettore come parametro ad una funzione, in realtà si sta passando l'indirizzo dell'elemento di indice **0**.
- ▶ Il parametro formale deve essere di tipo **puntatore** (al tipo degli elementi del vettore)
- ▶ di solito si passa anche la dimensione del vettore in un ulteriore parametro.

### Esempio:

```
void stampaVettore(int *v, int dim)
{ int i;
  for (i = 0; i < dim; i++)
    printf("v[%d]: %d\n", i, v[i]);
}
```

```
main()
{ int vet[5] = {1, 2, 3, 4, 5};
  ...
  stampaVettore(vet, 5);    ... }
```

- ▶ Per evidenziare che il parametro formale è un vettore (ovvero l'indirizzo dell'elemento di indice 0), si può utilizzare la notazione `nome-parametro[]` invece di `*nome-parametro`.

**Esempio:** `void stampa(int v[], int dim) { ... }`

- ▶ Si può anche specificare la dimensione nel parametro, ma questa viene ignorata.

**Esempio:** `void stampa(int v[5], int dim) { ... }`

- ▶ Come al solito, nel prototipo della funzione il nome del parametro (vettore) può anche mancare.

**Esempio:** `void stampa(int [], int);`

- ▶ Il passaggio di un vettore è un **passaggio per indirizzo**.  
⇒ La funzione può modificare gli elementi del vettore passato.

**Esempio:** Lettura di un vettore.

```
void leggiVettore(int v[], int dim)
{
    int i;
    for (i = 0; i < dim; i++)
    {
        printf("Immettere l'elemento di indice %d: ", i);
        scanf("%d", &v[i]);
    }
}
```

## Esempio: Programma che legge, inverte e stampa un vettore di interi

```
#include <stdio.h>
#define LUNG 5

void leggiVettore(int [], int);
void stampaVettore(int [], int);
void invertiVettore(int [], int);

main()
{
    int vett[LUNG];

    leggiVettore(vett, LUNG);
    printf("Vettore prima dell'inversione\n");
    stampaVettore(vett, LUNG);

    invertiVettore(vett, LUNG);
    printf("Vettore dopo l'inversione\n");
    stampaVettore(vett, LUNG);
}
```

- La definizione della procedura

`void invertiVettore(int [], int);` è lasciata per **esercizio**.