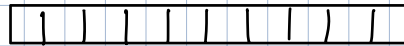


STRUTTURE  
DATI

# GLI ARRAY / I VETTORI

↳ ARRAY MONODIMENSIONALI

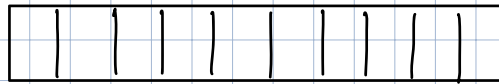


CONCETTUALMENTE COME:

UNA CASSETTIERA CON TANTI  
CASSETTI

CREARE UN VETTORE

`int a[10];`



DIMENSIONE = 10 CELLE

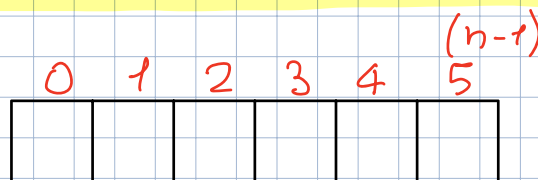
↳ OVERO: `int "nome vettore"[n];`

↑  
TIPO  
VARIABILI  
DELL'ARRAY

↑  
NOME DEL  
VETTORE

DIMENSIONE DEL  
VETTORE

N.B.



$n = 6$  DIMENSIONE DEL VETTORE!

## INIZIALIZZARE UN VETTORE

`int vet[5] = {1, 2, 3, 4, 5};`    vet

0	1	2	3	4
1	2	3	4	5

`int vet2[5] = {0};`    vet2

0	1	2	3	4
0	0	0	0	0

## ASSEGNARE / SOSTITUIRE DEI VALORI AD UN VETTORE

`int vet[5] = {1, 2, 3, 4, 5};`    vet

0	1	2	3	4
1	2	3	4	5

`vet[3] = 10;`    vet

0	1	2	3	4
1	2	3	10	5

~~`vet[5] = 2;`~~ NON SI PUO' FARE! VADO FUORI DAL VETTORE

## INPUT / OUTPUT SU UN VETTORE

```
#include <stdio.h>
```

```
#define dim 10
```

```
int main() {
```

```
    int vet[dim] = {0}; // INIZIALIZZO TUTTO A 0
```

```
    int i;
```

```
    for (i=0; i < dim; i++) {
```

```
        printf("... ");
```

```
        scanf("%d", &vet[i]);
```

INSERISCO UN  
VALORE NELLA  
i-ESIMA POSIZIONE  
DEL VETTORE

INPUT DA

TASTIERA DEI

VALORI PER IL

VETTORE

GUARDO  
TUTTE LE  
POSIZIONI  
DEL  
VETTORE

```
    }  
    for (i=0; i < dim; i++) {
```

```
        printf("%3d", vet[i]);
```

GUARDO LA  
POSIZIONE  
i-ESIMA DEL  
VETTORE

OUTPUT / STAMPA

DI UN VETTORE

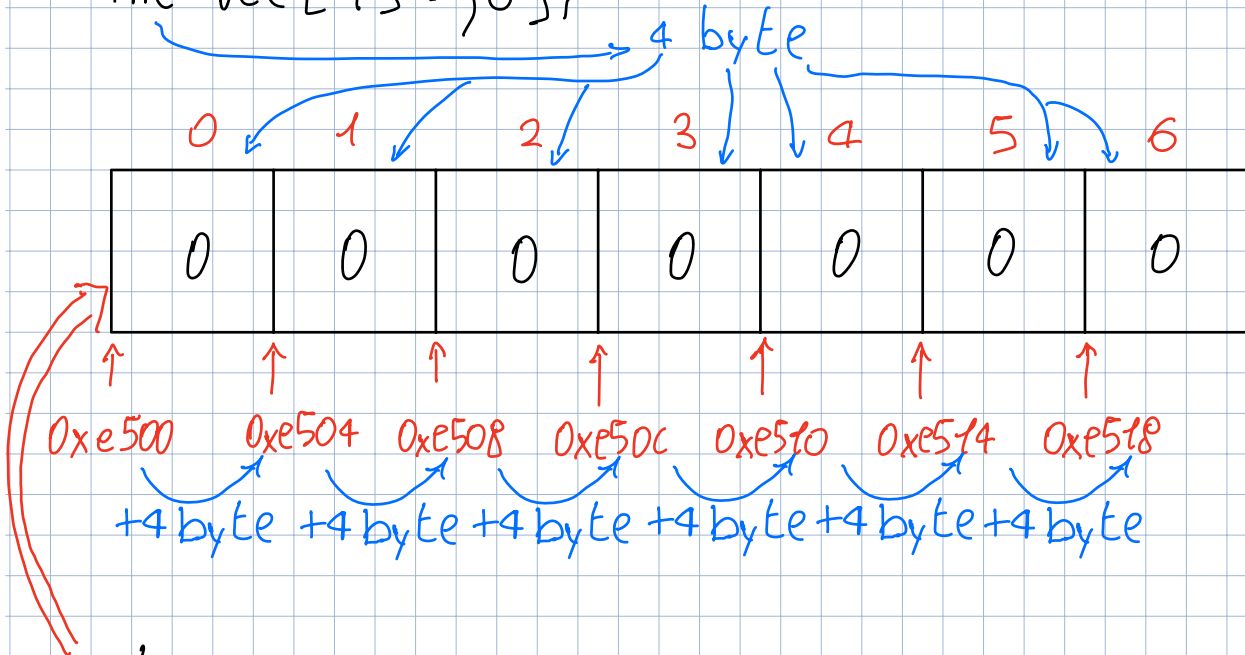
A VIDEO

```
    }
```

```
} // FINE MAIN()
```

## N.B. INDIRIZZI DEL VETTORE

```
int vet[7] = {0};
```



`vet`  $\Leftarrow$  NOME DEL VETTORE PUNTA ALLA MEMORIA DELLA  
PRIMA CELLA DEL VETTORE (CELLA 0)

DENTRO `vet` C'È QUINDI L'INDIRIZZO ALLA  
PRIMA CELLA DI MEMORIA DEL VETTORE

$\Rightarrow$  QUINDI IL NOME DEL VETTORE È UN PUNTATORE  
ALLA PRIMA CELLA DEL VETTORE

## VEETTORE DI CHAR => STRINGHE

```
char stringa[5] = {'c', 'i', 'a', 'o'};
```



↑  
VEETTORE DI 5 byte

'c'	'i'	'a'	'o'	'\0'
-----	-----	-----	-----	------

← '\0' = TERMINATORE DELLA

STRINGA (LO VEDREMO

MEGLIO CON LE STRINGHE)

## PUNTORI CON GLI ARRAY

```
int vet[5] = {1, 2, 3, 4, 5};
```

vet →

0	1	2	3	4
1	2	3	4	5

## vet PUNTA ALLA PRIMA CELLA DEL VETTORE

↳ QUINDI:

- printf("%d\n", \*vet); // COSA STAMPA? → 1

E:

- printf("%d\n", \*(vet + 3)); // COSA STAMPA? → 4

N.B. OGNI ELEMENTO / CELLA DEL VETTORE È A TUTTI GLI EFFETTI UNA VARIABILE SU CUI ESEGUIRE TUTTE LE OPERAZIONI POSSIBILI / VISTE IN PRECEDENZA.

ESEMPIO CODICE:

```
#include <stdio.h>
```

```
#define DIM 10
```

```
int main() {
```

```
    int vet[DIM] = { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 };
```

```
    int i;
```

```
    for (i = 0; i < DIM; i++) {
```

```
        vet[i] = vet[i] * 2;
```

```
    }
```

```
}
```