

11/12/2018

Lezione 5

Gli array vengono passati ai sottoprogrammi solo per indirizzo quindi ogni modifica fatta all'interno del sottoprogramma inerente all'array verrà effettuato in quello "originale", inoltre la dichiarazione del parametro formale non richiede di specificare il numero degli elementi che compongono gli array.

* Nella dichiarazione del prototipo, se si vuole passare un vettore, si mettono tante parentesi quadre quante sono le dimensioni dell'array in modo da specificare il tipo di dato al prototipo della funzione *

* Per rendere l'array imm modificabile bisogna precedere al tipo la parola "const", inoltre l'array non può essere 'ritornato' da una funzione *

* Per il passaggio di array multidimensionali, solo la prima dimensione si può non definire le altre invece sono obbligatorie *

Un record è una collezione di dati inerenti ad un unico oggetto, in C il record prende il nome di struct e la dichiarazione è la seguente:

```
typedef struct
{
    tipo1 var1;
    tipo2 var2;
    ...
} NomeStruct;
```

Per manipolare una singola componente del record si utilizzerà l'operatore ".", inoltre di default il passaggio di parametri ad un sottoprogramma avviene attraverso il passaggio di valore, infine si può anche ritornare una struct.

* Si può anche fare una copia dell'intero record*

*** Si utilizza la parentesi per evidenziare il nome della struct in modo da richiamare l'indirizzo del nome della struct e non l'indirizzo del campo

ES:

(*studente).nome → Va Bene, il puntatore evidenzia il nome della struct

*studente.nome → NON Va Bene, il puntatore punta a tutto) ***

TUTTO QUESTO VIENE SOSTITUITO DALL'OPERATORE → "OPERATORE DI SELEZIONE INDIRETTA"-