

# LINGUAGGIO C++

## Strutture dati: Record e Tabelle

*Prof. Armando Valentino*

# Record

Un **record** è un elenco di dati di tipo non omogenei.  
Ogni dato che compone il record si chiama **campo** ed è identificato con il nome

In C++ i record sono chiamati **Strutture** e si definiscono con l'istruzione **struct**

```
struct nomeStruttura{  
    tipo campo1;  
    tipo campo2;  
    .....  
    tipo campoN;  
};
```

obbligatorio ;

- Campi di tipo uguale si possono scrivere sullo stesso rigo
- Un campo può essere uno qualsiasi dei tipi semplici conosciuti, un array o una struttura
- La struttura è un tipo di dato

Per utilizzare la struttura si deve dichiarare una variabile del tipo della struttura definita:

Es:        *nomeStruttura nomevariabile;*

# Strutture

Per fare riferimento ad un campo della struttura si usa la **notazione puntata**.

nomevariabile.nomecampo  Individua il campo 'nomecampo' della struttura 'nomevariabile'

Le variabili struttura possono essere inizializzate durante la dichiarazione con i valori nello stesso ordine dei campi dichiarati nella struttura

**Es:** nomeStruttura nomevariabile = (valoreCampo1, ....., valoreCampoN);

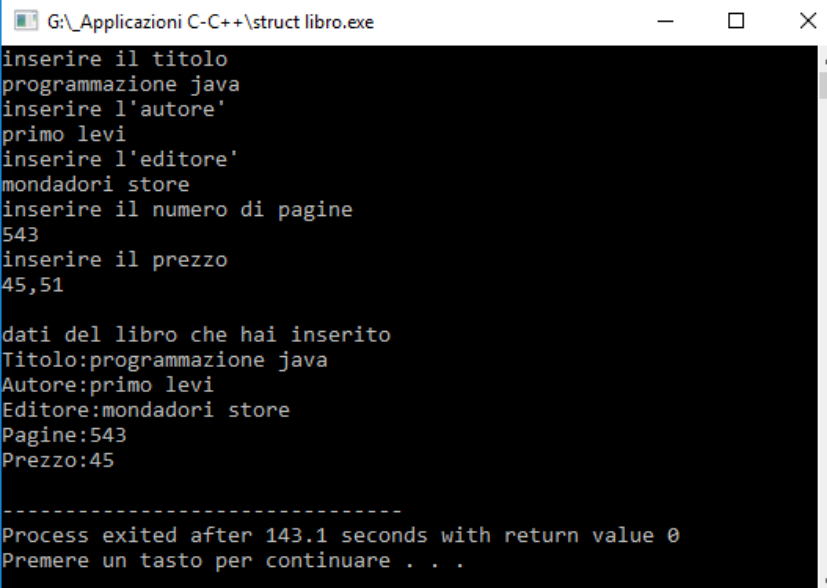
Esempio: struttura Libro

```
Struct tipoLibro{  
    string titolo, autore, editore;  
    int numpagine;  
    double prezzo;  
}
```

# Esempio Libro

```
#include <iostream>
using namespace std;
struct tipoLibro{
    string titolo, autore, editore;
    int numpagine;
    double prezzo;
};

int main()
{
    tipoLibro libro;
    cout << "inserire il titolo " << endl;
    getline(cin, libro.titolo);
    cout << "inserire l'autore " << endl;
    getline(cin, libro.autore);
    cout << "inserire l'editore " << endl;
    getline(cin, libro.editore);
    cout << "inserire il numero di pagine" << endl;
    cin >> libro.numpagine;
    cout << "inserire il prezzo" << endl;
    cin >> libro.prezzo;
    cout << endl;
    cout << "dati del libro che hai inserito" << endl;
    cout << "Titolo:" << libro.titolo << endl;
    cout << "Autore:" << libro.autore << endl;
    cout << "Editore:" << libro.editore << endl;
    cout << "Pagine:" << libro.numpagine << endl;
    cout << "Prezzo:" << libro.prezzo << endl;
    getchar();
    return 0;
}
```



```
G:\_Applicazioni C-C++\struct libro.exe

inserire il titolo
programmazione java
inserire l'autore'
primo levi
inserire l'editore'
mondadori store
inserire il numero di pagine
543
inserire il prezzo
45,51

dati del libro che hai inserito
Titolo:programmazione java
Autore:primo levi
Editore:mondadori store
Pagine:543
Prezzo:45

-----
Process exited after 143.1 seconds with return value 0
Premere un tasto per continuare . . .
```

# Tabelle

## Un array di record è detta TABELLA

Un array di record può esser rappresentato in forma tabellare utilizzando tante righe quanti sono i record e tante colonne quanti sono i campi

Esempio:

TITOLO	EDITORE	AUTORE	PAGINE	PREZZO
LINGUAGGIO C++	MONDADORI	MARIO ROSSI	420	25
INFORMATICA	HOEPLI	MARCO VERDI	624	30
STORIA D'ITALIA	ZANICHELLI	LUISA MARRO	250	12
PASCAL	MONDADORI	CARLO PIERI	300	20

[Esercizio Tabella libri](#)

# Esercizio Tabella Libri

**Scrivere un programma per realizzare una tabella di libri; utilizzare le funzioni per realizzare il programma:**

- funzione per inserire i dati di un libro da input
- funzione per visualizzare i dati di un libro
- funzione per inserire un libro in array
- funzione per stampare l'elenco dei libri
- funzione per stabilire quanti libri si vuole inserire

[Esercizio Tabella libri](#)

# Ordinamento tabella

Una tabella si dice **ordinata in base ai valori di un campo** (detto campo **chiave**) se i valori del campo sono in ordine crescente o decrescente.

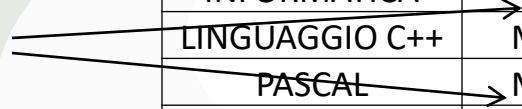
Una tabella ordinata su un campo chiave può avere diversi record con valori uguali del campo chiave

Ogni volta che il valore del campo chiave è diverso dal precedente si parla di **rottura del codice** (cambiamento del contenuto del campo)

TABELLA LIBRI ORDINATA SUL CAMPO EDITORE

TITOLO	EDITORE	AUTORE	PAGINE	PREZZO
INFORMATICA	HOEPLI	MARCO VERDI	624	30
LINGUAGGIO C++	MONDADORI	MARIO ROSSI	420	25
PASCAL	MONDADORI	CARLO PIERI	300	20
STORIA D'ITALIA	ZANICHELLI	LUISA MARRO	250	12

Rottura  
di codice



Alla **rottura del codice** si possono eseguire istruzioni quando **cominciano** o **quando terminano**.  
Ad esempio stampa di intestazioni o stampa di totali, medie, ec...

# Ordinamento Tabella

Scrivere una funzione che ordina la tabella libri sul campo Editore  
Utilizzare ordinamento insertion sort realizzando una funzione scambia

```
void scambia( tipoLibro &a, tipoLibro &b){
    tipoLibro c;
    c=a;
    a=b;
    b=c;
}
void Ordina(tipoLibro vetLibri[],int dim){
    for (int i=0; i<dim-1;i++) {
        for(int j=i+1; j<dim; j++){
            if (vetLibri[i].editore > vetLibri[j].editore) {
                scambia(vetLibri[i], vetLibri[j] );
            }
        }
    }
}
```

[Tabella Libri con ordinamento](#)



# Esercizio: Gara

Ad una gara atletica partecipano atleti nome, cognome, numero di pettorina. Ogni atleta realizza un tempo in secondi durante la gara.

Creare un programma in C++ che permette di:

1. Inserire i dati dei partecipanti e dei tempi realizzati
2. Stampare la classifica di tutti i concorrenti
3. Stampare i primi tre classificati con l'indicazione del vincitore della gara
4. Dato un numero di pettorina stampare il nome del concorrente e la posizione finale in classifica
5. Dato un tempo in secondi, stampare tutti gli atleti che hanno realizzato un tempo inferiore

Utilizzare le funzioni per realizzare l'applicazione.

# Esercizio: fantacalcio

Definire la struttura dati Squadra che descrive una squadra di calcio ideale con il relativo allenatore.

Una squadra è identificata da nome, colore della maglia, punteggio corrente classifica e nome allenatore. L'allenatore è una persona caratterizzata da nome e cognome più il numero dei titoli vinti (un numero intero).

Scrivere un programma C++ che, dichiarata una tabella Fantacalcio di tipo Squadra (per comodità se ne fissi la dimensione a 10 elementi), consenta di inserire da tastiera i dati delle squadre e dei relativi allenatori.

Operazioni:

1. visualizzare il solo cognome degli allenatori di squadre che hanno più di 30 punti in classifica
2. Numero totale di allenatori che hanno vinto almeno un titolo.
3. Stampare l'elenco delle squadre con allenatori che hanno un numero titoli vinti superiore a 3.

Utilizzare le funzioni per realizzare il programma

# Licenza



Quest'opera è distribuita con Licenza [Creative Commons Attribuzione - Non commerciale - Non opere derivate 4.0 Internazionale](https://creativecommons.org/licenses/by-nc-nd/4.0/).