

Stringhe e  
strutture

---

Che cosa  
imparerai a  
fare

Definire, confrontare e  
concatenare stringhe

Estrarre sottostringhe

Trattare le stringhe come array di  
caratteri

Dichiarare e usare le strutture  
come nuovo tipo di dato

Che cosa  
dovrai  
studiare

Sintassi del  
metodo length

Operazione di  
somma tra  
stringhe

Sintassi per  
estrarre una  
sottostringa

Operazioni di  
confronto tra  
stringhe

Concetto di  
dato  
aggregato

Concetto di  
struttura e la  
sua sintassi

Stringa

Sequenza di  
caratteri, racchiusa  
tra virgolette  
doppie

# Dichiarazione di stringa

```
#include <string>
```

```
...
```

```
string nome = "giovanni";
```

```
...
```

```
nome = "carlo";
```

```
...
```

```
cout << "inserisci il tuo nome: " ;
```

```
cin >> nome;
```

# Separatori di stringhe

Se avessi inserito "Mario Bianchi", il cin avrebbe preso solo "Mario"

Le parole sono divise da spazi, tabulazioni e vai a capo

Se voglio leggere anche Bianchi, devo usare un secondo cin su un'altra variabile:

- `cin >> nome;`
- `cin >> cognome;`

Per prendere tutto in una volta

`Getline(cin,nome)`

Legge tutti i caratteri  
digitati fino a che si  
preme INVIO e li  
salva in nome

Calcolare la lunghezza della  
stringa

```
Int n =  
nome_stringa.length();
```



# Concatenazione

Usando l'operatore '+' tra stringhe

```
string nome = "Mario";
```

```
string cognome = "Bianchi"
```

```
string totale = nome + cognome;
```

# Sottostringhe

```
nome_stringa.substr(inizio, lunghezza)
```

s è la stringa a cui viene applicata la funzione substr

inizio è il numero di carattere da cui si inizia ad estrarre

lunghezza è quanti caratteri voglio prendere

# Esempio

```
string saluto_mondo =  
    "Hello, World!";
```



```
string saluto =  
    saluto_mondo.substr(0,5);
```

# Confronti tra stringhe

OPERATORE	DESCRIZIONE	USO
==	Uguale a	Lettere minuscole e maiuscole sono diverse
<	Minore di	Seguono l'ordinamento alfabetico: prima i numeri, poi le lettere maiuscole e infine le lettere minuscole
>	Maggiore di	
<=	Minore o uguale di	
>=	Maggiore o uguale a	
!=	Diverso	

## Esempi

"Harry"=="HARRY"

"Tom"<"Dick"

"Tom"<"dick"

"auto"<"automa"

"Harry">5

Stringa come  
char[]

```
String nome = "Mario";
```

```
Char[] nome = new  
char[] {'M','a','r','i','o'};
```

# Struttura

Tipi di dati aggreganti, cioè  
capaci di contenere tipi di  
dati diversi

```
Struct nome_struttura
```

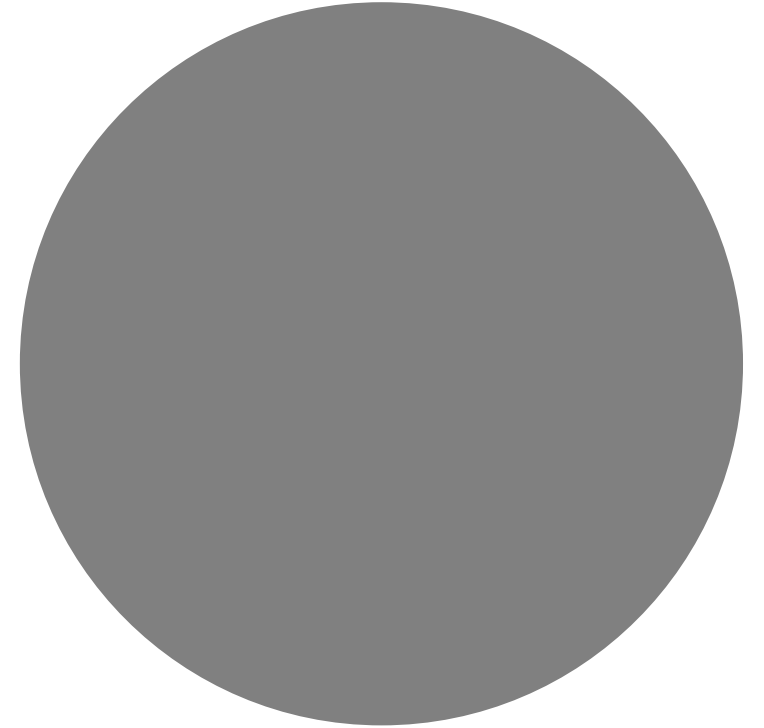
```
{
```

```
//membri della struttura
```

```
}
```

---

Sintassi struttura





# Membri della struttura

Non sono altro che  
variabili

Utilizzate per gestire  
quantità di dati non  
molto grandi e che  
occupano poca  
memoria

```
struct studente{  
    string nome;  
    string cognome;  
}
```

Esempio

Esempio  
dichiarazione  
e  
riempimento

```
studente alunno1;
```

```
alunno1.nome = "Leonardo";
```

```
alunno1.cognome = "Rizzon";
```

Funzione  
costruttore  
della struct

```
struct studente{  
    string nome;  
    string cognome;  
  
    public studente(string nome, string cognome){  
        nome = nome;  
        cognome = cognome;  
    }  
}
```

```
studente alunno2("Emma", "Berlato");
```

LEARN TO  
CODE