

# WEB DEVELOPER

## Fondamenti di Programmazione

### Massimo PAPA

# Le stringhe in C++

## - Funzioni predefinite -

## Funzioni Predefinite

### ● Funzioni Predefinite sulle Stringhe

- `s.size()`: calcola la lunghezza della stringa
- `s.insert(x, s1)`: inserimento di una stringa in un'altra
- `s.erase(x, y)`: cancellazione di una sottostringa
- `s.substr(x, y)`: estrazione di una sottostringa
- `s.find(s1)`: ricerca della posizione di una sottostringa
- `s.replace(x, y, s1)`: sostituzione di una sottostringa con un'altra
- `s.c_str()`: cambia la rappresentazione di un valore da string ad una stringa del C

## Funzioni Predefinite

- Lunghezza: `s.size()`
  - calcola la lunghezza della stringa `s`
- Esempi:

```
string s = "*****";
```

```
cout << s.size();
```

12

```
string cognome;
```

```
cout << cognome.size();
```

0

## Funzioni Predefinite

- Inserimento: `s.insert(x, s1)`
  - inserisce la stringa `s1` nella stringa `s` prima della posizione `x`

- Esempi

```
string s = "Mario Rossi";  
cout << s.insert(10, "in");
```

Mario Rossini

## Funzioni Predefinite

- Cancellazione: `s.erase(x, y)`
  - cancella `x` caratteri dalla stringa `s` a partire dalla posizione `y`
- Esempio

```
string s0 = "Mario Rossi";  
cout << s0.erase(3, 3);
```

MarRossi

## Funzioni Predefinite

- Estrazione di sottostringa: `s.substr(x,y)`
  - restituisce la sottostringa della stringa `s` fatta di `y` caratteri a partire dalla posizione `x`
- Esempi:

```
string s = "Mario Rossi";  
string p1 = s.substr(6, 5);  
cout << p1;
```

Rossi

## Funzioni Predefinite

- Ricerca di sottostringa: `s.find(s1)`
  - restituisce la posizione della sottostringa `s1`
- Esempio

```
string s0 = "Alex Del Piero";  
cout << s0.find("el");
```

6

## Funzioni Predefinite

- Sostituzione: `s.replace(x, y, s1)`
  - sostituisce nella stringa `s` `y` caratteri a partire dalla posizione `x` con la stringa `s1`
- Esempi:

```
string s = "Mario Rossi";  
cout << s.replace(3,3,"ta e Tina ");
```

Marta e Tina Rossi



## Funzioni Predefinite

- Trasformazione in stringa C: `c_str()`
  - trasforma un valore di tipo `string` nel corrispondente valore di tipo “stringa C”
  - indispensabile per usare funzioni e procedure che prevedono argomenti che sono stringhe C
- Esempio: creazione di un flusso su file

```
string nomeFile;  
cin >> nomeFile;  
ifstream flussoDaFile(nomeFile.c_str());
```