



ELEMENTI DI INFORMATICA

DOCENTE: FRANCESCO MARRA

INGEGNERIA CHIMICA

INGEGNERIA ELETTRICA

SCIENZE ED INGEGNERIA DEI MATERIALI

INGEGNERIA GESTIONALE DELLA LOGISTICA E DELLA PRODUZIONE

INGEGNERIA NAVALE

UNIVERSITÀ DEGLI STUDI DI NAPOLI FEDERICO II
SCUOLA POLITECNICA E DELLE SCIENZE DI BASE

INPUT DI STRINGHE



AGENDA

- Input di stringhe
 - `cin.ignore()`
 - `cin.get()`
 - `cin.getline()`

PROPRIETÀ DELLA LETTURA DALLO STDIN

- Nel caso di estrazione dallo stdin, la lettura dei dati non è *sincronizzata* rispetto alla loro digitazione
 - I caratteri vengono preliminarmente accumulati in un *buffer di memoria*
 - Se all'atto dell'esecuzione di *cin* il buffer è vuoto, il programma si ferma in attesa che i dati siano digitati finché non si invia un *Carriage Return* (CR)
 - Caricamento del buffer e prelievo dei dati si susseguono finché viene assegnato un valore a tutte le variabili
 - Eventuali dati che restano nel buffer di input dopo che l'operazione di lettura è terminata vengono utilizzati nella istruzione di lettura successiva

PROPRIETÀ DELLA LETTURA DALLO STDIN

- La lettura di un dato dal buffer termina quando si incontra un blank, un carattere di tabulazione o un CR
- Se il dato da leggere è di tipo *numerico*, la lettura termina quando si incontra un carattere non valido
 - L'errore di digitazione non genera messaggi di errore
 - Non si assegna valore alla variabile
 - La condizione di errore inibisce anche le altre operazioni di lettura
- Se il dato da leggere è di tipo *char*, viene letto un solo carattere
- Un effetto collaterale del funzionamento di cin è l'*effetto eco*
 - I caratteri digitati da tastiera vengono anche visualizzati sullo schermo

ESEMPI DI INSERIMENTO VALORI

```
std::cout << "Inserire valore prima variabile: ";  
std::cin >> var1;  
std::cout << "Inserire valore seconda variabile: ";  
std::cin >> var2;  
std::cout << "Inserire valore terza variabile: ";  
std::cin >> var3;
```

Es. 1: Inserimento valori man mano che vengono richiesti

```
Inserire primo valore: 10  
Inserire secondo valore: 20  
Inserire terzo valore: 30
```

Es. 2: Inserimento valori in una sola volta

```
Inserire primo valore: 10 20 30  
Inserire secondo valore:  
Inserire terzo valore:
```

LETTURA DELLE STRINGHE

FUNZIONE CIN.GET

- La funzione *cin* non può essere sempre usata per la lettura delle stringhe di caratteri
 - Considera lo spazio come un separatore
- La funzione ***cin.get*** consente di prelevare dal buffer un numero massimo di caratteri, compreso spazio e CR
 - È spesso usata per realizzare funzioni di inserimento controllato
 - Il carattere CR è sostituito dalla funzione con il terminatore nullo '\0'

```
cin.get(nome_stringa, num_max_car);
```

LETTURA DELLE STRINGHE

FUNZIONE CIN.GETLINE

- La funzione **cin.getline** consente di specificare anche il carattere di terminazione

- Per default, il carattere di terminazione è il CR
- Anche in questo caso il carattere di terminazione è sostituito da '\0'
- Se il carattere di terminazione non è CR, i CR vengono inclusi nella stringa

```
cin.getline(nome_stringa, num_max_car, car_fine);
```

- Esempio:

- Carattere di terminazione uguale al dollaro

```
cin.getline(linea, 25, '$');
```


GESTIONE DEGLI ERRORI

FUNZIONI CIN.IGNORE

- Si faccia attenzione che **cin.get** e **cin.getline** lasciano il carattere CR nel buffer dello stream di input, per cui una successiva lettura non va a buon fine in quanto incontra il carattere CR lasciato dalla prima lettura
- Prima di eseguire **cin.get** o **cin.getline** bisogna eliminare dal buffer i caratteri CR eventualmente presenti a causa di una lettura precedente
- Ciò può essere fatto, ad esempio, con il metodo **cin.ignore()**

ESEMPIO: CONFRONTO TRA DUE STRINGHE

```
#include<string.h>
#include<iostream>
#define lunghezza_massima 30
using namespace std;

int main(){
    char parola1[lunghezza_massima];
    cout<<"Inserisci la prima parola: ";
    cin.get(parola1,lunghezza_massima);

    char parola2[lunghezza_massima];
    cout<<"Inserisci la seconda parola: ";
    cin.ignore(); //svuoto il buffer
    cin.get(parola2,lunghezza_massima);

    if( strcmp(parola1,parola2)==0 )
        cout<<"Le due parole sono uguali."<<endl;
    else
        cout<<"Le due parole non sono uguali."<<endl;
}
```

ESERCIZIO 1

- Scrivere un programma che, acquisita una stringa composta da più parole, permetta all'utente di scegliere quale delle seguenti operazioni eseguire:
 - Visualizzazione della stringa
 - Visualizzazione della stringa in maiuscolo
 - Visualizzazione della stringa senza gli spazi
 - Visualizzazione della stringa invertita
 - Visualizzazione del numero di caratteri della stringa
 - Visualizzazione del numero di parole della stringa
 - Visualizzazione della ricorrenza di un carattere nella stringa

ESERCIZIO 2

- Realizziamo un programma che consenta di inserire delle parole (anche con spazi), e poi stamparle ordinate

DOMANDE, DUBBI, PERPLESSITÀ

