



UNIVERSITY OF TRENTO - Italy

# Laboratorio 2

**Pierluigi Roberti**  
**Carmelo Ferrante**

DISI – aa 2025-2026

Università degli Studi di Trento

[pierluigi.roberti@unitn.it](mailto:pierluigi.roberti@unitn.it)

# Lettura singolo dato da tastiera

Una volta dichiarata una variabile intera

```
int dato;
```

Per poter valorizzare la variabile con un valore proveniente da tastiera usare (**C** like):

```
scanf("%d", &dato);
```

**NB &dato significa indirizzo della variabile dato**

Per scrivere il contenuto della variabile intera usare:

```
printf("valore = %d", dato);
```

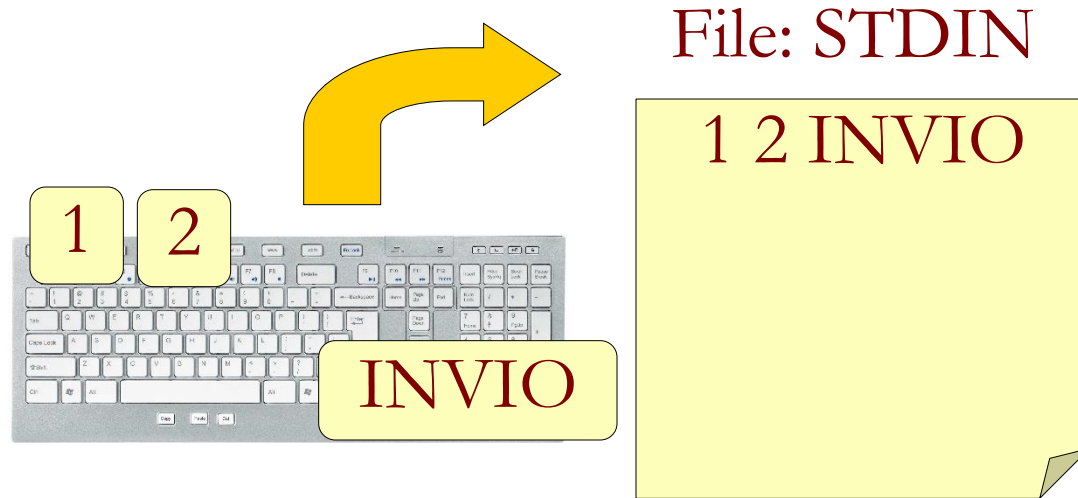
Similmente per le variabili di tipo reale float (%f) e di tipo carattere char (%c)

Approccio **C++**

```
cin >> dato;
```

```
cout << dato;
```

# Lettura di più dati da tastiera

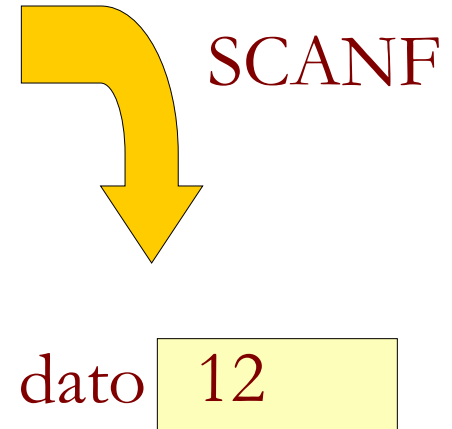
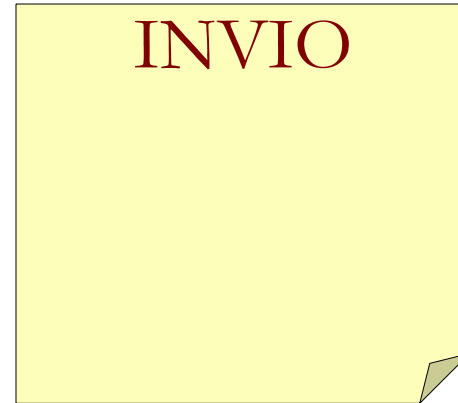


Cosa viene “letto” dal sistema dopo l’input di un numero intero: **`scanf(“%d”, &dato);`**

Supponiamo che utente digiti il numero 12 e che confermi con INVIO

# Lettura di più dati da tastiera

File: STDIN

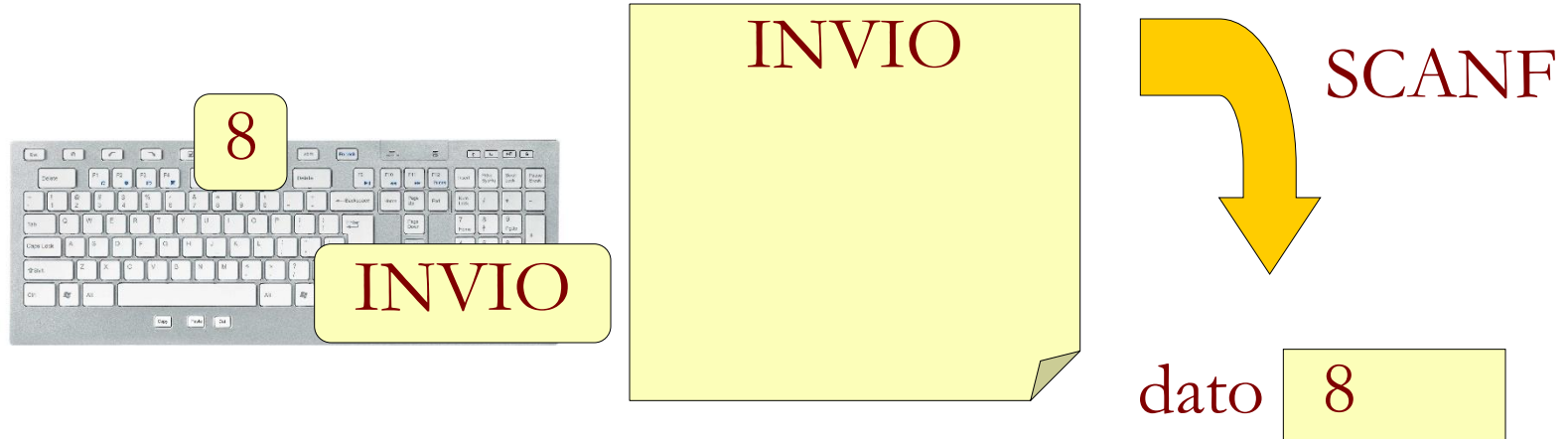


Il solo valore numerico 12 è inserito nell'indirizzo della variabile dato.

Nel file STDIN rimane un valore pari al tasto INVIO.

# Lettura di più dati da tastiera

File: STDIN

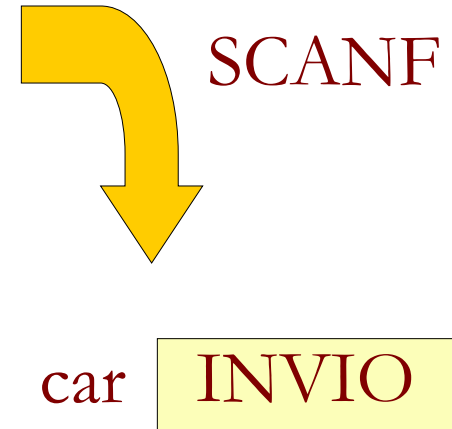
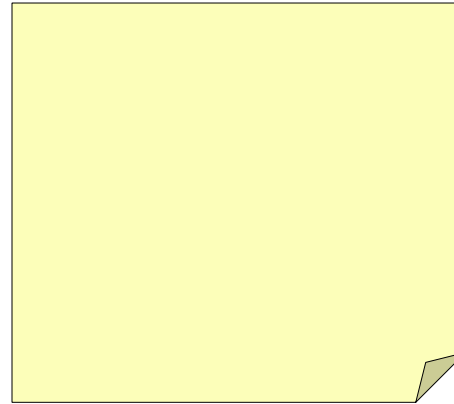


A questo punto se volessi valorizzare un secondo dato di tipo numerico (int o float) il valore presente nel file STDIN (INVIO) non sarebbe considerato dallo **scanf** perchè **dato non numerico scanf(“%d”, &dato);**

Supponiamo che utente digiti il numero 8 e che confermi con INVIO, come prima il valore 8 è inserito nell'indirizzo della variabile dato.

# Lettura di più dati da tastiera

File: STDIN



Se però ora volessi valorizzare una variabile di tipo carattere:

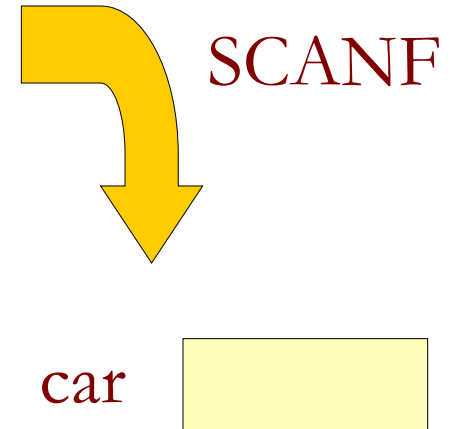
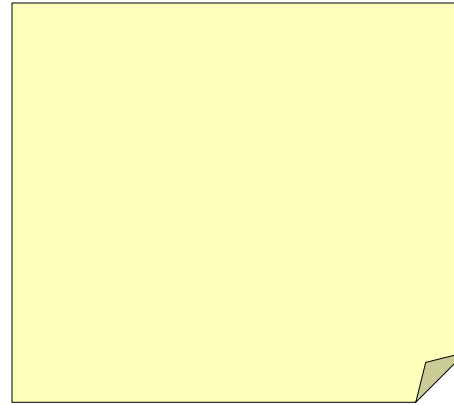
```
char car;
```

Tramite una **scanf("%c",&car);**

verrebbe prelevato dal file di input STDIN il valore INVIO, in quanto valore valido per una variabile di tipo char.

# Lettura di più dati da tastiera

File: STDIN



Quindi è **NECESSARIO SEMPRE** svuotare il file STDIN prima di leggere una variabile di tipo carattere (char)

Tramite l'istruzione

```
fflush(stdin); //per s.o. windows
```

```
fpurge(stdin); //per s.o. mac
```

**Prima** della istruzione **scanf("%c",&car);**