

Input e Output di Base

Input/Output

- L'immissione dei dati di un programma e l'uscita dei suoi risultati avvengono attraverso operazioni di **lettura e scrittura**
- Il C **non ha istruzioni predefinite** per l'input/output
- In ogni versione ANSI C, esiste una *Libreria Standard* (**stdio.h**) che mette a disposizione alcune funzioni (dette *funzioni di libreria*) per effettuare l'input e l'output

stdio.h

- Le dichiarazioni delle funzioni messe a disposizione da tale libreria devono essere incluse nel programma:

#include <stdio.h>

- **#include** è una direttiva per il **preprocessore C**:
- nella fase precedente alla compilazione del programma ogni direttiva “#...” viene eseguita, provocando delle modifiche testuali al programma sorgente.
- Nel caso di **#include <nomefile>** viene sostituita l’istruzione stessa con il contenuto del file specificato
- **Dispositivi standard di input e di output:**
 - per ogni macchina, sono periferiche predefinite (generalmente tastiera e video)

Scrittura con formato: **printf**

- **printf** scrive una serie di valori in base alle specifiche contenute in *<stringa-formato>*
- I valori visualizzati sono i risultati delle espressioni che compaiono come argomenti
- **printf** restituisce il numero di caratteri scritti (il valore è molto spesso ignorato)
- La stringa di formato della **printf** può contenere sequenze costanti di caratteri da visualizzare

```
printf (<stringa-formato>, arg1, arg2, ...);
```

printf: specifiche di formato

- **Formati più comuni**

int %d oppure %i

unsigned int %u

esadecimale %x

float o double %f %e %g

carattere singolo %c

stringa di caratteri %s

- **Caratteri di controllo:**

newline \n

tab \t

- NB; Per la stampa del carattere ' % ' si usa: %% per la stampa di \ si usa \\

Lettura con formato: **scanf**

- **scanf** legge una serie di valori in base alle specifiche contenute in *<stringa-formato>* e memorizza i valori letti nelle variabili
- restituisce il **numero di valori letti** e memorizzati, oppure EOF in caso di *end of file*
- Gli **identificatori** delle variabili a cui assegnare i valori sono sempre preceduti dal **simbolo &** (ne parleremo diffusamente...)
- La *<stringa_formato>* può contenere dei caratteri qualsiasi (che vengono scartati, durante la lettura), che si prevede vengano immessi dall'esterno, insieme ai dati da leggere
 - ➔ Esempio: `scanf ("%d:%d:%d", &A, &B, &C);`
richiede che i tre dati da leggere vengano immessi separati dal carattere ":"

Lettura con formato: **scanf**

- **scanf()** assegna i valori letti alle variabili specificate come argomenti (nell'ordine di lettura)

Ad esempio:

```
int x;  
float y;  
scanf("%d %f", &x, &y);
```

IMPORTANTISSIMO: Le specifiche di formato sono praticamente uguali a quelle della printf. L'unica eccezione è la lettura di double che necessita `%lf` anziché `%f`

```
double x;  
scanf("%lf", &x);
```

Esempio

```
main() {  
    int k;  
    scanf("%d", &k);  
    printf("Quadrato di %d: %d", k, k*k);  
}
```

- Se in ingresso viene immesso il dato **3** esso viene letto tramite la `scanf` e assegnato a **k**
- La **printf** stampera':
Quadrato di 3: 9