

03. Variabili, Tipi Primitivi e Costrutti Condizionali

Fondamenti di Programmazione e Laboratorio

Chiara Bodei, Damiano Di Francesco Maesa, Roberta Gori

CdL in Matematica, Università di Pisa
a.a. 2021/2022

Approfondimento: scanf I

Nella stringa di formato della scanf non mettete caratteri che non siano specificatori di formato: niente spazi, niente a capo.

```
scanf("%d%d", &base, &altezza}
```

Approfondimento: scanf II

Attenzione a leggere i caratteri ! Se scrivete

```
scanf ("%c%c", &c1, &c2}
```

allora in input dovete darli attaccati

ab

Se invece volete separarli da uno spazio avete due modi

```
scanf ("%c%*c%c", &c1, &c2}
```

oppure

```
scanf ("%c%c%c", &c1, &dummy, &c2}
```

Approfondimento: uso del Cast I

Ogni dato deve avere il tipo corretto.

Se una base di un triangolo e' un intero va dichiarata come int anche se poi potreste avere un'operazione che la coinvolge nella quale volete che sia un reale.

Due motivi per dichiararla intero:

- motivo logico: il tipo e' anche un controllo sui valori possibili che aiuta il debugging, se dichiarate base come float non avete piu' nessun controllo sui valori ammissibili.
- motivo pratico: la rappresentazione di un float puo' introdurre errori di approssimazione che gli int non hanno.

Quindi, quando serve, introdurre un **cast** secondo la sintassi:

```
(tipo) espressione;
```

Approfondimento: uso del Cast II

L'operatore di cast ha **precedenza più alta** degli operatori binari e **associa da destra a sinistra**

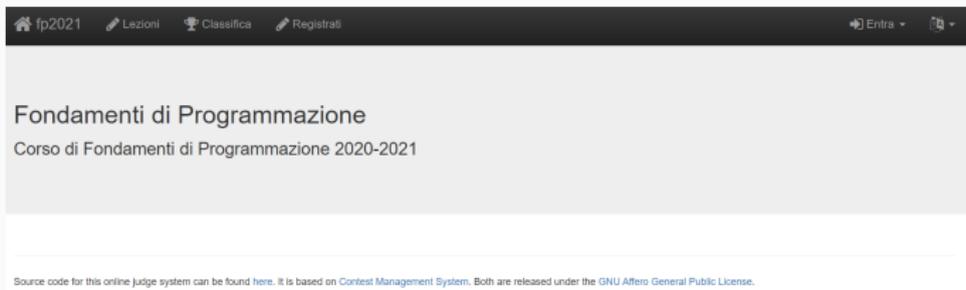
Esempio

```
int somma, n;  
float media;  
media = somma/n; /* divisione tra interi */  
media = (float)somma/n; /* divisione tra reali */
```

Nell'esempio, `(float)somma/n` equivale a `((float)somma)/n`

Piattaforma di autovalutazione

N.B. Gli Screenshots sono della piattaforma dello scorso anno ma tranne che per fp2021 invece che fp2022 il resto e' tutto uguale!



`http://fp2122.dijkstra.di.unipi.it`

fp2021 Lezioni Classifica Registrati Entra

Dati di accesso al sito

Username

Password

Ripeti password

Dati personali

Nome

Cognome

Indirizzo e-mail

Ripeti e-mail

Registrati

Anteprima del profilo utente



(username)
(Nome) (Cognome)

Due raccomandazioni:

- usate un **username** che sia riconducibile a voi
(es. nome.cognome)
- non scordate la **password** :)



Le varie esercitazioni sono

- organizzate per lezione,
- a partire dalla *Lezione 3* (quella odierna)

Ogni esercizio è spiegato in un PDF, accessibile dalla scheda **Testo**

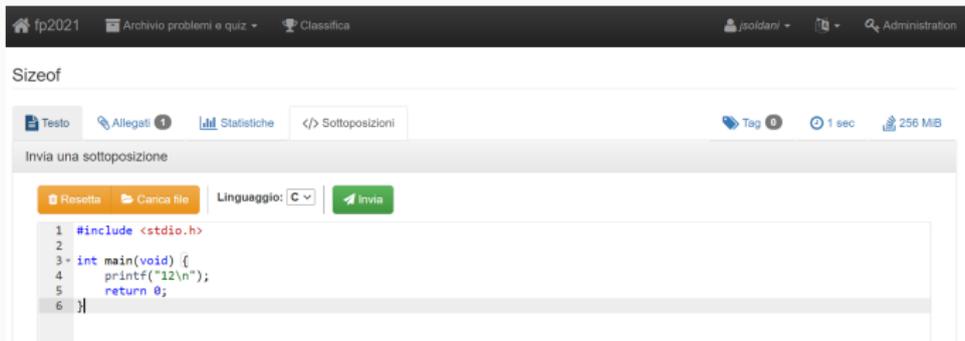


The screenshot shows a web browser window with a dark header bar containing navigation links: 'fp2021', 'Archivio problemi e quiz', 'Classifica', 'jsolidani', and 'Administration'. Below the header, the page title is 'Sizeof'. A navigation bar includes 'Testo' (selected), 'Allegati 1', 'Statistiche', and 'Sottoposizioni'. On the right, there are icons for 'Tag 0', '1 sec', and '256 MB'. The main content area displays the title 'Sizeof' and the section 'Esercizio' with the text: 'Scrivere un programma che legga da tastiera un numero intero x e stabilisca il numero di byte necessario a memorizzare x variabili di tipo intero.' Below this is the 'Esempio' section, which contains a table with two columns: 'Input' and 'Output'. The table shows the input '3' and the corresponding output '12'.

Input	Output
3	12

Esercizi: Sottomissione

La soluzione può essere inviata dalla scheda **Sottomissione** (caricando direttamente il file .c o copiando il contenuto nel form)



Cliccando su **Invia** il sistema valuta la vostra soluzione

Esercizi: Valutazione

Esito della valutazione:

- se corretta, il sistema vi assegna il punteggio 100/100 (massimo),
- altrimenti vi segnala errore o assegna un punteggio inferiore

Cliccando sull'ID della sottomissione potete conoscere su quali test case il vostro programma ha fallito

The screenshot shows a submission details window titled "Dettagli della sottoposizione". The window contains the following information:

Testcase	Risultato	Dettagli	Tempo	Memoria
000	Correct	Output is correct	0.002s	240 KiB
001	Not correct	Output isn't correct	0.002s	240 KiB
002	Not correct	Output isn't correct	0.002s	240 KiB
003	Not correct	Output isn't correct	0.002s	240 KiB

Output della compilazione

Esito della compilazione:	ok
Tempo di compilazione:	0.188s
Memoria utilizzata:	15.0 MiB

Standard output

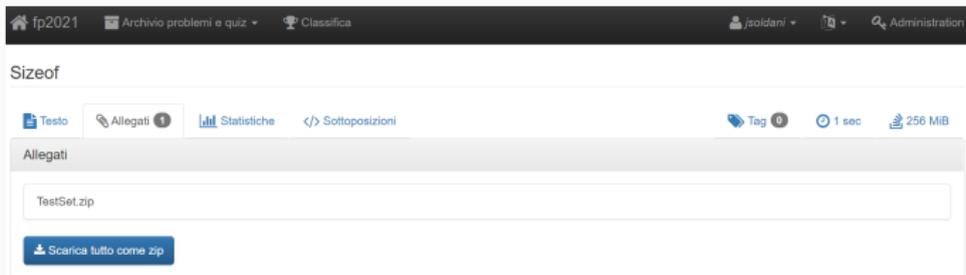
Standard error

File

Scarica

Esercizi: Test Case

Trovate i test case considerati nell'esercizio nella scheda **Allegati**, inseriti in un file ZIP scaricabile



Ogni test case N è formato da una coppia

- `inputN.txt` (input fornito)
- `outputN.txt` (output atteso)

Esercizi: Test Case (2)

Se non avete ottenuto il massimo punteggio, è perchè uno di questi casi fallisce. Dovete “debuggare” il vostro programma e capire perchè non fornisce l’output atteso

```
./nomeEseguibile < inputN.txt
```

consente di lanciare il vostro programma eseguibile fornendogli in input quanto contenuto nel file `inputN.txt`

```
./nomeEseguibile < inputN.txt | diff - outputN.txt
```

idem, e inoltre verifica quali siano le differenze tra l’output generato dal vostro programma e quello atteso (contenuto in `outputN.txt`)

Due cose:

- ricordatevi di terminare il main dei vostri programmi C con `return 0`

Due cose:

- ricordatevi di terminare il main dei vostri programmi C con `return 0`
- cercate di rimanere ai vertici della classifica :)
- alla fine cimentatevi in un esercizio divertente (prossima slide)!



The screenshot shows a web interface for a competition. At the top, there is a dark navigation bar with the text "fp2021", "Archivio problemi e quiz", "Classifica", and a user profile "jsoldani". Below this is a pagination control with numbers 1, 2, 3, 4, 5 and navigation arrows. The main content area displays a list of participants:

	asdrubale.agasti (Asdrubale Agasti)	★ 0
	pdor.figliodikmer (Pdor Figlio di Kmer)	★ 0

Buon Lavoro!

Esercizio Opzionale - Optional1Lab3 su Replit

Nella morra due giocatori si sfidano scegliendo un simbolo ciascuno tra sasso, forbici e carta: due simboli uguali pareggiano, mentre il sasso batte le forbici, le forbici battono la carta e la carta batte il sasso.

Scrivere un programma c che gestisca una sfida tra PC e utente:

- (a) generando un numero casuale da 1 a 3 (1=sasso, 2=forbici, 3=carta, utilizzare il costrutto `#define` per rendere leggibile l'associazione tra il numero e il simbolo),
- (b) leggendo un carattere ('s'=sasso, 'f'=forbici, 'c'=carta) e
- (c) stampando l'esito del confronto. Se l'utente immette un carattere diverso da 's', 'f' e 'c' allora perde comunque.

Come generiamo numeri casuali nella prossima pagina.

Esercizio Opzionale - Optional1Lab3 su Replit

Per generare un numero pseudo-casuale compreso tra 1 e 3 utilizzare il seguente codice

```
#include <time.h>
#include <stdlib.h>

int    pc_simbolo;
/* Per sapere cosa fanno le funzioni srand() e rand()
** usare il comando linux 'man srand' */
srand(time(NULL));
pc_simbolo=(rand()%3)+1;
```

la variabile pc_simbolo conterrà un numero pseudo-casuale tra 1 e 3 (inclusi).