



Fondamenti di Informatica

Laurea in

Ingegneria Civile e Ingegneria per l'ambiente e il territorio

Linguaggio C: Le basi

Stefano Cagnoni e Monica Mordonini

Le funzioni di input/output

<stdio.h>

Input / Output

- Il linguaggio C consente di definire *funzioni* aggiuntive raggruppabili eventualmente in *librerie*.
- Esiste in particolare una libreria standard (*standard library*) di funzioni; per rendere disponibili tali funzioni, occorre inserire nelle dichiarazioni globali del programma, la direttiva di precompilazione:

```
#include <stdio.h>
```

- La libreria standard comprende, tra le altre, le seguenti funzioni di input/output

Input / Output

```
printf (<stringa di formato> [<lista di espressioni>])
```

- per visualizzare dati sullo *standard output*
- **<lista di espressioni> ::= <espressione> { <espressione> }**
 - sono espressioni il cui valore deve essere visualizzato
- **<stringa di formato>**: contiene i caratteri da visualizzare ed i simboli di formato per i valori delle espressioni da visualizzare:
 - %d per la visualizzazione di un numero intero
 - %f per la visualizzazione di un numero reale
 - %c per la visualizzazione di un carattere
 - %s per la visualizzazione di una stringa

Esempi

```
printf ("Questa è una frase di prova\n");
```

- Questa è una frase di prova

```
printf ("La somma di %d e %d risulta: %d\n", m, n, m+n);
```

- La somma di 3 e 2 risulta: 5
(se la variabile m ha valore 3 e la variabile n ha valore 2)

```
printf ("Il codice ASCII %d\nviene visualizzato come %c\n", 61, 61)
```

- Il codice ASCII 61
viene visualizzato come A

Input / Output

```
scanf (<stringa di formato> , <lista indirizzi  
variabili>)
```

- consente la lettura di dati dallo *standard input*
- **<lista di indirizzi di variabili> ::= <indirizzo> { <indirizzo> }**
<indirizzo> ::= &<variabile>
- **<stringa di formato>** (vedi printf)

- Esempi
scanf ("%d %f %c", &n, &x, &c);
Ma
scanf ("%s", nomestringa); (una variabile
stringa contiene l'indirizzo del suo primo carattere)

Input / Output

Attenzione

- scanf e printf hanno sintassi simili.
- Ma scanf è di input ed ha bisogno di sapere l'indirizzo della cella in cui voglio memorizzare il dato d'ingresso
- Quindi ogni variabile (scalare) deve essere preceduta dal simbolo &
- printf è una funzione di output ed ha bisogno di sapere il valore della variabile e quindi non serve &.

Il linguaggio C

7

Esercizi

Scrivere un programma che prenda in ingresso un codice (un numero intero) e ne visualizzi il carattere (char) cui è associato nella tabella ASCII

Scrivere un programma che prenda in ingresso un carattere e ne visualizzi il codice (numero intero) a cui è associato nella tabella ASCII

Il linguaggio C

8

Byte	Code	Char	Byte	Code	Char	Byte	Code	Char	Byte	Code	Char
00000000	0	Nul	00100000	32	Sp	01000000	64	@	01100000	96	a
00000001	1	Start of heading	00100001	33	!	01000001	65	A	01100001	97	b
00000010	2	Start of text	00100010	34	"	01000010	66	B	01100010	98	c
00000011	3	End of text	00100011	35	#	01000011	67	C	01100011	99	d
00000100	4	End of transmit	00100100	36	\$	01000100	68	D	01100100	100	e
00000101	5	Enquiry	00100101	37	%	01000101	69	E	01100101	101	f
00000110	6	Acknowledge	00100110	38	&	01000110	70	F	01100110	102	g
00000111	7	Audible bell	00100111	39	'	01000111	71	G	01100111	103	h
00001000	8	Backspace	00101000	40	(01001000	72	H	01101000	104	i
00001001	9	Horizontal tab	00101001	41)	01001001	73	I	01101001	105	j
00001010	10	Line feed	00101010	42	*	01001010	74	J	01101010	106	k
00001011	11	Vertical tab	00101011	43	+	01001011	75	K	01101011	107	l
00001100	12	Form Feed	00101100	44	,	01001100	76	L	01101100	108	m
00001101	13	Carriage return	00101101	45	-	01001101	77	M	01101101	109	n
00001110	14	Shift out	00101110	46	.	01001110	78	N	01101110	110	o
00001111	15	Shift in	00101111	47	/	01001111	79	O	01101111	111	p
00010000	16	Data link escape	00110000	48	0	01010000	80	P	01110000	112	q
00010001	17	Device control 1	00110001	49	1	01010001	81	Q	01110001	113	r
00010010	18	Device control 2	00110010	50	2	01010010	82	R	01110010	114	s
00010011	19	Device control 3	00110011	51	3	01010011	83	S	01110011	115	t
00010100	20	Device control 4	00110100	52	4	01010100	84	T	01110100	116	u
00010101	21	Neg. acknowledge	00110101	53	5	01010101	85	U	01110101	117	v
00010110	22	Synchronous idle	00110110	54	6	01010110	86	V	01110110	118	w
00010111	23	End trans. block	00110111	55	7	01010111	87	W	01110111	119	x
00011000	24	Cancel	00111000	56	8	01011000	88	X	01111000	120	y
00011001	25	End of medium	00111001	57	9	01011001	89	Y	01111001	121	z
00011010	26	Substitution	00111010	58	:	01011010	90	Z	01111010	122	{
00011011	27	Escape	00111011	59	;	01011011	91	[01111011	123	
00011100	28	File separator	00111100	60	<	01011100	92	\	01111100	124]
00011101	29	Group separator	00111101	61	=	01011101	93]	01111101	125	^
00011110	30	Record Separator	00111110	62	>	01011110	94	^	01111110	126	_
00011111	31	Unit separator	00111111	63	?	01011111	95	_	01111111	127	Del

Il linguaggio C

9

Esempio

- Legge un codice ASCII e ne visualizza il carattere

```
#include <stdio.h>
int main(){
    char c;

    printf ("Inserire il codice ASCII:");
    scanf ("%d", &c);
    printf ("Il carattere corr. è %c\n", c);
    return 0;
}
```

Il linguaggio C

10

Strutture di controllo

If, while, for, ...

Strutture di controllo

Selezione if

if (<condizione>) <istruzioni> [else <istruzioni>]
Se la condizione è vera esegue il primo blocco di istruzioni, altrimenti esegue il secondo (se specificato)

<istruzioni> ::= <istruzione> | '{' <istruzione> '{' <istruzione> }'

esempio
if (x > 0)
 y = x;
else
 y = -x;
printf("dato %d\n", y);

falso ↔ 0 vero ↔ non 0	operatori di confronto maggiore > maggiore o uguale >= minore < minore o uguale <= uguale == diverso !=
operatori logici not ! and && or	

Il linguaggio C

12

Esempio

```
/* Dati due numeri in ingresso, individua il maggiore */
#include <stdio.h>
int main(){
    int num1,num2;

    printf("Immetti i due numeri:");
    scanf("%d%d",&num1,&num2);
    if (num1==num2)
        printf("I due numeri sono uguali");
    else
        if(num1>num2)printf("Il primo è maggiore");
        else printf("Il secondo è maggiore");
    return 0;
}
```

Esercizio: Fare il diagramma di flusso di questo problema

Strutture di controllo

■ Ciclo for

```
for(<istr.iniziale>; <condizione>; <istr. ciclica>)
{<istruzioni>}
```

Dopo avere eseguito l'istruzione iniziale, ripete l'esecuzione del blocco di istruzioni che segue finché la condizione resta vera (se è falsa la prima volta, può anche non eseguirlo mai), concludendo ogni ciclo o con l'esecuzione dell'istruzione ciclica.

← esempio: calcolo della somma dei primi 100 numeri naturali

```
#include <stdio.h>
int main(){
    int i, somma = 0;

    for ( i=1; i<=100; i=i+1)
        somma = somma+i; /* se c'è una sola istruzione si
                           possono tralasciare le graffe */
    printf ("La somma vale: %d\n", somma);
    return 0;}
```

Strutture di controllo

■ Ciclo while

```
while (<condizione>) {<istruzioni>}
```

Esegue il blocco di istruzioni che segue, finché la condizione è vera (come per il for, può anche non eseguirlo mai)

← esempio: calcolo della somma dei primi 100 numeri naturali

```
#include <stdio.h>
int main(){
    int i=1;
    int somma = 0;

    while(i<=100) {
        somma = somma+i;
        i=i+1;
    }
    printf ("La somma vale: %d\n", somma);
    return 0;}
```

Strutture di controllo

■ Ciclo do while

```
do <istruzioni> while (<condizione>)
```

Esegue (sempre, almeno una volta) le istruzioni comprese nel blocco che segue e le ripete finché la condizione è vera

← esempio: calcolo della somma dei primi 100 numeri naturali

```
#include <stdio.h>
int main(){
    int i=1;
    int somma = 0;

    do {
        somma = somma+i;
        i=i+1;
    } while (i<=100)
    printf ("La somma vale: %d\n", somma);
    return 0;}
```

Esercizi

Scrivere un programma che prenda in ingresso un intero N e calcoli il doppio della somma dei primi N numeri

Scrivere un programma che chieda in ingresso un numero finché non si inserisce un numero dispari. Quando questo avviene, il programma deve uscire scrivendo quanti numeri pari erano stati inseriti in precedenza.