

Manuali di C

- Linguaggio C
B. W. Kernigan, D. M. Ritchie
Jackson
- La guida completa al C
H. Schildt
McGraw-Hill

Fondamenti di Informatica

1

Linguaggio C

- Creato nei primi anni 70 per sviluppare il S.O. Unix
- È un linguaggio di programmazione di uso generale
- ANSI C (1989)

Fondamenti di Informatica

2

Sintassi del C

- Alfabeto:
 - lettere maiuscole e minuscole (A-Z, a-z)
 - cifre decimali (0-9)
 - caratteri speciali +-*/=<>()[]{}.,;:~^"?!%#&|
- Vocabolario
 - parole chiave (int, if, unsigned, while,...)
 - identificatori
 - valori
- Regole sintattiche e di punteggiatura

Fondamenti di Informatica

3

Struttura di un programma

```
#include <stdio.h>

main()
{
  /* dichiarazioni */
  int dato;

  /* istruzioni */
  dato = 4;
  printf("dato = %d\n", dato);
}
```

Fondamenti di Informatica

4

Identificatori

- Nomi usati dal programmatore per identificare variabili, costanti, funzioni, ...
- Composti da lettere, numeri e underscore; iniziano con una lettera
- Unici nel campo di validità (primi 31 caratteri)
- Scope e linkage

Fondamenti di Informatica

5

Tipi di dati

- Definisce l'insieme dei valori e l'insieme delle operazioni
- Tipi base:
 - **char** (byte con segno, -128 -> 127)
 - **int** (interi con segno, 16bit: -32768 -> 32767, 32bit: -2.147.483.648 -> 2.147.483.647)
 - **float, double** (reali, 32bit: -3.4E38 -> 3.4E38, 64bit: -1.7E308 -> 1.7E308)
- Qualificatori: short, long, signed, unsigned

Fondamenti di Informatica

6

Convenzioni per i valori

- **Interi**: numeri con o senza segno, senza punto decimale
 - prefissi: 0x (esadecimale), 0 (ottale)
 - suffissi: l oppure L (long), u oppure U (unsigned)
- **Reali**: numeri con o senza segno, con punto decimale o notazione scientifica
 - suffissi: f oppure F (float), l oppure L (long)
- **Caratteri**: caratteri racchiusi tra apici
 - sequenze di escape

Fondamenti di Informatica

7

Variabili e costanti

- Dichiarazione delle variabili
tipo identificatore {, identificatore};
- Dichiarazione delle costanti
const *tipo identificatore = valore costante;*
#define *identificatore testo*

Fondamenti di Informatica

8

Operatori

- operatore di assegnamento
- operatori aritmetici
- operatori di manipolazione dei bit
- operatori di incremento e decremento
- operatori relazionali
- operatori logici

Fondamenti di Informatica

9

Operatore di assegnamento

variabile = espressione;

L'assegnamento è un'espressione; il suo valore è quello della variabile dopo l'assegnamento.

Esempio:

variabile1 = variabile2 = espressione;

Fondamenti di Informatica

10

Operatori aritmetici

Agiscono su dati di tipo intero e reale

* / %
+ -

- regole di priorità
- conversioni di tipo
- scrittura abbreviata

somma = somma + dato; somma += dato;

Fondamenti di Informatica

11

Operatori di manipolazione dei bit

Agiscono su dati di tipo intero e carattere

- & AND
- | OR
- ^ EXOR
- ~ NOT
- << left shift
- >> right shift

Fondamenti di Informatica

12

Operatori di incremento e decremento

Agiscono su tutti i tipi di dato

++ --

dato++; **dato = dato + 1;**
dato--; **dato = dato - 1;**

- Possono essere prefissi o suffissi

Fondamenti di Informatica

13

Operatori relazionali

Agiscono su espressioni aritmetiche

< > <= >=
== !=

- Il risultato di un'espressione relazionale è un valore logico (1 per vero, 0 per falso)

Fondamenti di Informatica

14

Operatori logici

Agiscono su dati di tipo logico (nell'accezione del C) e espressioni logiche

! (NOT)
&& (AND) || (OR)

- Il risultato è un valore logico!

Fondamenti di Informatica

15

Ingresso e uscita dei dati

- I periferici sono visti come un flusso continuo di dati verso il programma (input) o dal programma (output)
- Non esistono istruzioni di I/O, si usano funzioni di libreria

#include <stdio.h>

Fondamenti di Informatica

16

La funzione printf

printf("format", arg1, arg2, ..., argn);

- visualizzazione su video (stdout)
- formato = testo + specificatori di formato
 - per il tipo **int**: **%d, %u %o %x %c**
 - per il tipo **double**: **%f, %e**
- argomenti: variabili, costanti, espressioni di vari tipi
- lo specificatore di formato può comprendere:
 - flag, larghezza, precisione, modificatori

Fondamenti di Informatica

17

La funzione scanf

scanf("format", arg1, arg2, ..., argn);

- lettura da tastiera (stdin)
- formato = testo + specificatori di formato
- argomenti: variabili di vari tipi precedute da "&"
- lo specificatore di formato può comprendere:
 - modificatori, larghezza, *
- comportamento per lettura di numeri e caratteri

Fondamenti di Informatica

18
