

## Le variabili

Come leggere dati da tastiera,  
fare un'elaborazione e  
visualizzare il risultato

1

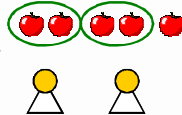
## Un programma un po' più elaborato

- Finora abbiamo imparato a visualizzare sullo schermo una frase, o un insieme di caratteri (stringa)
- Però ci interessa anche prendere in ingresso dei dati che poi il calcolatore possa elaborare.
- Ci interessa usare l'elaboratore per elaborare dati di ogni tipo (numeri, parole, suoni, immagini, ...)
- Dobbiamo quindi essere in grado di rappresentare i dati all'interno del sistema

2

## Algoritmo: calcolo divisione intera

- Supponiamo di voler calcolare la divisione fra due numeri interi, con quoziente e resto
- Istruzioni base:
  - so calcolare il quoziente dati il dividendo ed il divisore: operazione /
  - so calcolare il resto dati il dividendo ed il divisore: operazione %
- Algoritmo:
  - Chiedi all'utente il **dividendo** e mettilo in una cella di memoria
  - Chiedi all'utente il **divisore** e mettilo in una cella di memoria
  - calcola il **quoziente = dividendo / divisore** e mettilo in un'altra cella
  - calcola il **resto = dividendo % divisore** e mettilo in un'altra cella
  - visualizza **quoziente** e **resto**



3

## Variabili

- Ci serve avere a disposizione delle celle di memoria in cui mettere i valori inseriti dall'utente, calcolare dei risultati, ecc.
- Per questo, nel linguaggio C ci sono le **variabili**:
  - sono delle **astrazioni** di celle di memoria:
    - nelle celle di memoria possiamo mettere solo sequenze di 0 e 1,
    - mentre noi vorremmo avere una visione un po' più astratta: vogliamo, ad esempio, metterci dei numeri interi, o frazionari, ...

4

## Definizione della variabile

- Per creare una variabile, in C bisogna prima definirla con questa sintassi:  
`<definizione variabile> ::= <tipo> <identificatore>;`
- L' `<identificatore>` è un nome che decidiamo noi
- Il `<tipo>` ci dice
  - quali **operazioni** possiamo fare con quel dato:
    - con un numero potremo fare somme, sottrazioni, ...
    - con un'immagine, una parola o un suono potremo fare operazioni diverse
  - quali **valori** possiamo assegnarvi
  - come è **rappresentato** internamente il dato
- Es:  

```
int a;
```

crea una variabile di tipo `int` (intero) che si chiama `a`, con cui potremo fare operazioni di somma, sottrazione, divisione fra interi, ...
- Da questo momento possiamo usare la variabile, inserendo dei valori ed usandola per effettuare elaborazioni

5

## INIZIALIZZAZIONE DI UNA VARIABILE

- **Contestualmente alla definizione è possibile specificare un valore iniziale per una variabile**
- **Inizializzazione di una variabile:**  
`<tipo> <identificatore> = <espr>;`
- **Esempio:**  

```
int x = 32;
```

6

## Assegnamento di valori

- Per assegnare un valore ad una variabile si usa il simbolo =
- Questo simbolo rappresenta l'assegnamento distruttivo: il valore che c'era prima viene cancellato e viene inserito il nuovo valore
- $a = 7;$
- $a = 5;$
- alle variabili si può anche assegnare il risultato di espressioni
- $a = a + 1;$
- $b = (a + 3) / 2;$

a 

a 

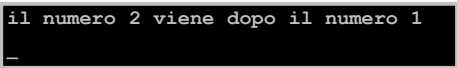
7

## Calcolo di una divisione intera

```
#include <stdio.h>
main()
{ int a;           // creazione delle variabili che mi servono
  int b;
  int c;
  int d;
  a = 5;          // assegno il valore iniziale
  b = 2;
  c = a / b;      // quoziente della divisione intera
  d = a % b;      // resto della divisione intera
  printf("il quoziente e': %d\n il resto e': %d",c,d); // stampo il risultato
}
```

8

## Stampa di numeri interi

- Possiamo, con la `printf`, stampare anche numeri interi:  
`printf("Il numero %d viene dopo il numero %d\n",2,1);`  
stampa la frase:  

- `%d` vuol dire: prendi il prossimo argomento; pensalo come un intero e visualizzalo in decimale (in base 10)

9

## Calcolo dell'area di un rettangolo

```
#include <stdio.h>
main()
{ int a;
  int b;
  int c;
  a = 7;
  b = 3;
  c = a*b;
  printf("%d", c);
}
```

l'asterisco è il simbolo della moltiplicazione in C

10

## Quali nomi dare alle variabili?

- **identificatori:** sequenze di caratteri tali che  
`<Identificatore> ::= <Lettera> { <Lettera> | <Cifra> }`  
`<Lettera> ::= 'A' .. 'Z', 'a' .. 'z'`  
`<Cifra> ::= '0' .. '9'`  
Intuitivamente un identificatore è una sequenza (di lunghezza maggiore o uguale a 1) di lettere e cifre che inizia obbligatoriamente con una lettera. (No lettere accentate).
- Quindi come nomi possiamo utilizzare  
`a, b, aBc, x1, ...`  
però è importante rendere leggibile il programma, quindi cerchiamo di dare nomi significativi:  
`dividendo, resto, MediaEsami, ...`
- Nota che il C è "case sensitive": `abc` è diverso da `ABC`, da `Abc`, ...

11

## Calcolo di una divisione intera

```
#include <stdio.h>
main()
{ int dividendo, divisore, quoziente, resto;
  dividendo = 5;
  divisore = 2;
  quoziente = dividendo / divisore;
  resto = dividendo % divisore;
  printf("il quoziente e': %d\n",quoziente);
  printf("il resto e': %d",resto);
}
```

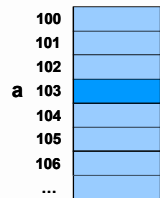
versione compatta:  
posso mettere tutte le dichiarazioni di variabili dello stesso tipo nella stessa riga, separando le variabili con la virgola

12

## Che cosa fa il compilatore?

```
int a;
a = 10;
a = a + 1;
```

- stabilisce un indirizzo in memoria dove mettere la variabile a, ad es, l'indirizzo 103

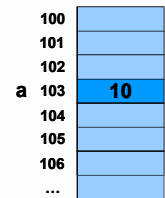


13

## Che cosa fa il compilatore?

```
int a;
a = 10;
a = a + 1;      mov (103), 10
```

- Aggiunge l'istruzione macchina
  - "metti il valore 10 nella cella 103"



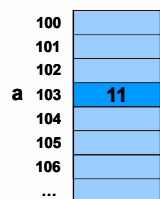
14

## Che cosa fa il compilatore?

```
int a;
a = 10;
a = a + 1;
```

- Aggiunge le istruzioni macchina
  - porta nella CPU il valore della cella 103
  - aggiungi 1
  - metti il risultato nella cella 103

```
mov (103), 10
mov AX, (103)
add AX, 1
mov (103), AX
```

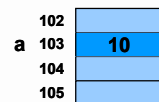


15

## Significato

- quindi nell'istruzione

$a = a + 1;$   
 la "a" di sinistra vuol dire "l'indirizzo di memoria della variabile a" (103)  
 la "a" di destra vuol dire "il valore della variabile a" (10)



16

## L-value

- Una variabile è un'astrazione della cella di memoria.
- Formalmente, è un simbolo associato a un indirizzo fisico (L-value)...

simbolo	indirizzo
x	1328

Perciò, l' L-value di x è 1328 (fisso e immutabile!).

17

## VARIABILI

... che denota un valore (R-value).

...	
1328	4
...	

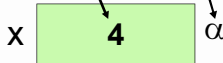
...e l' R-value di x è attualmente 4 (può cambiare).

18

## VARIABILI NEI LINGUAGGI IMPERATIVI

### Una variabile in un linguaggio imperativo

- **non è un sinonimo per un dato** come in matematica
- **è un'astrazione della cella di memoria**
- **associata a due diverse informazioni:**
  - **il contenuto (R-value)**
  - **l'indirizzo a cui si trova (L-value)**



19

## Inserimento di dati da tastiera

- Scrivere un programma che calcola la divisione fra 5 e 2 non è molto utile ... vorremmo che fosse l'utente ad inserire il **valore** delle variabili
- L'istruzione `scanf` serve per far sì che l'utente possa inserire dei dati da tastiera.
- Per ora, utilizziamo questa sintassi semplificata (vedremo più avanti altre potenzialità). Per leggere un intero ed inserirlo in una variabile `x`

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

- `%d` ha lo stesso significato usato nella `printf`: mi dice di che tipo è il dato (decimale, cioè in base 10).
- Vedremo più avanti perché ci vuole la `&`; intuitivamente, qui dobbiamo usare l'L-value della variabile: il significato di `scanf` è "leggi un valore e mettilo nella cella di indirizzo...". Dobbiamo quindi darle l'indirizzo (L-value) della variabile e non il suo valore (R-value).

20

## Programma completo

```
#include <stdio.h>
main()
{ int dividendo, divisore, quoziente, resto;
  printf("immetti il dividendo: "); // stampo un messaggio per l'utente
  scanf("%d",&dividendo); // leggo il valore del dividendo da tastiera
  printf("immetti il divisore: "); // stampo un messaggio per l'utente
  scanf("%d", &divisore); // leggo il valore del divisore da tastiera
  quoziente = dividendo / divisore;
  resto = dividendo % divisore;
  printf("il quoziente e': %d\n", quoziente);
  printf("il resto e': %d\n", resto);
}
```

Compila

21

## Operatori aritmetici fra interi

Operazione	numero argomenti	Simbolo
inversione di segno	1	-
somma	2	+
differenza	2	-
moltiplicazione	2	*
divisione	2	/
modulo (resto)	2	%

22

## Calcoliamo!!!

- A questo punto, se abbiamo una formula risolutiva, possiamo effettuare vari tipi di calcolo
  - Calcolare la somma di due numeri
  - Data una temperatura in gradi Celsius, trasformarla in gradi Fahrenheit ( $c * 9/5 = f - 32$ )
  - Dati  $a$  e  $b$ , calcolare la soluzione dell'equazione  $ax+b=0$
  - Calcolare l'area e il perimetro di un rettangolo dati la base e l'altezza
  - Letti da tastiera tre numeri che rappresentano le risposte di un sondaggio (numero di "sì", numero di "no", numero di "non so"), dire qual è la percentuale di "sì", di "no", di "non so"
  - ...

23