Introduzione ai vettori

© 2010 Fabio Proietti
Creative Commons
Attribution-ShareAlike 3.0 Unported
License

I tipi di dati in C

- int
- char
- float
- double
- Sono tutti tipi semplici

Definizione di vettore

- Per definire un vettore bisogna specificarne la dimensione, sia in modo diretto,
  - `int v[5];`
- ...che in modo indiretto
  - `int v[] = {3, 4, 2, 11, 9};`
  - `char w[] = "miao";`

Assegnazioni sui vettori

- Si accede ad ogni elemento specificando il valore dell'indice tra parentesi quadrate:
  - `v[3] = 0;`
  - `w[0] = 'z';`
- Attenzione: non c'è controllo sul limite superiore...
  - `v[3] = v[1] + v[8];`

Stringhe e vettori

- Un vettore di tipo **char** viene anche chiamato “stringa” e viene fatta terminare sempre con un valore nullo che in ottale si indica con /0 (equivale in binario a 00000000). Non confonderlo con il carattere '0' o con l'intero 0.
- `char w[] = "miao";`
- `char w[] = {'m','i','a','o','/0'}`
- “miao” è un vettore di 5 elementi!!!

posizione	0	1	2	3	4	5	
w	**m**	**i**	**a**	**o**	**/0**		

Input di stringhe

- Il segnaposto di una stringa è `%s`
- Scrivere l'identificatore del vettore equivale a scrivere l'indirizzo del suo primo elemento w[0]
  - `w` equivale a `&w[0]`
- Quindi con la funzione `scanf()` <u>non</u> si deve usare l'operatore "indirizzo di" (&)
  - `scanf("%s", w);`
- Eccetto nel caso in cui si legga un solo elemento
  - `scanf("%c", &w[2]);`

Output di stringhe

- Per usare `printf()` non ci sono differenze da ricordare
  - `printf("%s",w);`

Passaggio a funzioni

- Per il motivo già visto nell'uso della funzione **`scanf()`**, quando si passa un vettore ad una funzione, in realtà si sta passando il suo indirizzo
  - I parametri effettivi sono modificati!

Multidimensione

- int v[3][2];
  - È un vettore di 3 elementi ognuno dei quali è a sua volta di 2 elementi
- Un vettore bidimensionale può essere rappresentato sotto forma di tabella ma in memoria gli elementi sono disposti in modo lineare, facendo scorrere prima l'indice più <u>esterno</u>
  - v[0][0] = 0;
  - v[0][1] = 1;
  - v[1][0] = 2;
  - v[1][1] = 3;

0	2	
1	3	

0	1
2	3