

Array come parametri delle funzioni C

Quando un parametro di una funzione è un array, nell'intestazione della funzione non è necessario indicare la dimensione (ovvero il numero di elementi) dell'array. Infatti ciò che viene passato alla funzione è ciò che l'identificatore dell'array rappresenta, ovvero sia un *puntatore al primo elemento dell'array stesso*. Indicare la dimensione dell'array non aggiungerebbe nulla a tale informazione¹. Ad esempio la dichiarazione di funzione

```
int FunzioneA(int ParametroArray[])
```

è del tutto equivalente a quella

```
int FunzioneA(int ParametroArray[DIM_1])
```

dove DIM_1 è il numero di elementi dell'array. Un'altra forma equivalente alle precedenti è questa:

```
int FunzioneA(int *ParametroArray)
```

(la cosa appare ovvia considerando che l'identificativo di un array è un puntatore al primo elemento dell'array).

Quando uno dei parametri di una funzione è un array multidimensionale, il C richiede che di tale array siano indicate **tutte le dimensioni eccettuata la prima** (che è ininfluente e opzionale, esattamente come nel caso di array monodimensionali). Ad esempio la dichiarazione di funzione

```
int FunzioneB(int ParametroArrayMultidim[][DIM_2][DIM_3])
```

è del tutto equivalente a quella

```
int FunzioneB(int ParametroArrayMultidim[DIM_1][DIM_2][DIM_3])
```

Per spiegare perché è necessario indicare tutte le dimensioni eccetto (eventualmente) la prima di un parametro di tipo array è sufficiente considerare il fatto che in C l'identificativo di un array corrisponde ad un puntatore al primo elemento, e l'unico modo che il compilatore ha per raggiungere gli elementi successivi è di "saltare in avanti" nella memoria RAM di un numero di byte corrispondente a quelli richiesti da ciascun elemento dell'array. Nel caso di un array A i cui elementi sono a loro volta array (o array di array di...), è evidente che il numero di byte richiesti da ciascun elemento di A *dipende dal numero di elementi che compongono ciascuno degli array coinvolti nella struttura interna del singolo elemento di A*. Pertanto, quando A viene passato ad una funzione come parametro, è necessario specificare tutte queste dimensioni; resta invece opzionale specificare la dimensione di A.

¹ Qualora la dimensione di un parametro costituito da un array venga indicata (pur non essendo richiesta), essa non viene utilizzata dal compilatore. Resta dunque una responsabilità del programmatore fare in modo che la funzione non vada a leggere o scrivere fuori dai limiti dell'area di memoria occupata dall'array, esattamente come avviene in generale con gli array in C.