

## Il programmatore che c'è in noi - Lezione 2 - DATI

Nel precedente articolo ho dato le indicazioni relative a come "misurare" lo spazio occupato nella memoria del calcolatore utilizzando una nomenclatura non del tutto corretta come è stato giustamente suggerito da Paolo Portaluri. Dal link <http://www.germinara.it/iMemory.zip> potete scaricare un progetto in Cocoa (contenente sia l'applicazione sia i sorgenti) dove sono meglio chiarite le nomenclature. Il realtà tale progetto contiene anche le indicazioni relative all'argomento di oggi.

### I TIPI DI DATI BASE

La volta scorsa ho semplicemente fatto una distinzione tra  
DATI DI TIPO NUMERICO  
DATI DI TIPO CARATTERE  
DATI DI TIPO LOGICO

In realtà le cose sono un po' più complicate, il motivo di tale complicazione è dovuta al fatto che la memoria Ram (la nostra famosa scatola delle scarpe) non è grande all'infinito e quindi più riusciamo a ottimizzarne lo spazio meglio è'.

Bene, avete vinto al superenalotto e quindi cambiare casa, andate in un supermercato e prendete un po' di scatoloni di diverse dimensioni.

A questo punto, a casa, dovete mettere la vostra roba, nei vari scatoloni. Cosa fate ? Cercate di ottimizzare l'uso delle scatole a disposizione riempiendole con gli oggetti più indicati o per dirlo al contrario scegliete lo scatolone adatto al tipo di oggetto che dovrà contenere.

Ad esempio, non metterete un solo libro in uno scatolone enorme, oppure non strapperete un libro pur di farlo entrare in uno scatolone troppo piccolo.

Bene, anche per la nostra Memoria Ram (e disco fisso ecc.) dobbiamo fare necessariamente una ottimizzazione.

Questa ottimizzazione è possibile decidendo il tipo di dato idoneo per memorizzarne il futuro valore , se ad esempio devo memorizzare l'età di una persona, mi sarà sufficiente usare un tipo di dati che consenta di contenere un valore tra 0 e 120, di conseguenza si tratta di un numero INTERO, in questo caso SENZA SEGNO (non ha senso dire che una persona ha -5 anni...)

Se invece devo memorizzare quanti secondi sono passati da quando sono nato, ho bisogno di uno scatolone più grosso (in quanto il numero da memorizzare sarà notevolmente più grande).

Se devo memorizzare (=mettere nello scatolone) il valore di temperatura invernale in montagna probabilmente avrò necessità di indicare un valore tra - 25 e 30, usando in questo caso anche il numero negativo (quindi il segno).

Per farla breve, se devo memorizzare il costo del pieno della mia auto avrò necessità di indicare un valore REALE ossia con i numeri decimali (es. 63,58 ).

Tutto questo in "informaticese" si traduce in Tipi di Dati base i quali occupano una determinata dimensione nella nostra Ram ma potranno contenere solo determinati valori sia come tipo

(carattere, intero, reale, booleano) sia come LIMITI (ossia il valore minimo che posso far stare in quello scatolone ed il valore massimo).

Il tipo di dati piu' semplice è il char, che puo' contenere un solo CARATTERE o un numero che a seconda se con o senza segno, potra' essere compreso tra -128 e +127 oppure tra 0 e 255

1 char occupa 1 BYTE nella nostra memoria Ram

Questo significa che se devo memorizzare l'eta' di una persona potrei farlo usando un dato di tipo char in quanto il valore è compreso tra -128 e 127, è chiaro che se una persona ha 130 anni (beato lui...) non potremmo memorizzare la sua eta'.

Se ad esempio devo memorizzare l'iniziale del mio nome, allora va bene un char perche' e' un carattere (nel mio caso la F)

Davanti al nome del tipo di dato, puo' comparire un "modificatore" del tipo, che e' una ulteriore parola da scrivere prima del nome del tipo e specifica ulteriori informazioni per il tipo di dato, ad esempio se e' senza segno (unsigned) o con segno (signed).

Tipi di dati base

Nome	Dimensione occupata	Descrizione
char	- 1 BYTE	Carattere o numero intero piccolo
int	- 4 BYTES	Numero intero
float	- 4 BYTES	Numero reale (decimali) singola precisione
double	- 8 BYTES	Numero reale (decimali) doppia precisione

La dimensione occupata è indicativa, puo' variare in funzione del computer utilizzato.  
Per conoscere l'esatta dimensione occupata in memoria dal tipo di dato in linguaggio C si utilizza l'operatore sizeof(nomedeltipodidato)

Quindi

sizeof(char) mi dira' quanto spazio occupa un char in memoria  
sizeof(double) mi dira' quanto spazio occupa un double in memoria

e cosi' via.

Modificatori

Nome	Descrizione
signed	Il numero intero è con segno
unsigned	In numero intero è senza segno
long	Il numero è molto grande (lungo)
short	Il numero è piu' piccolo del normale (corto)

Esempi

unsigned char = vuol dire che puo' contenere un valore tra 0 e 255, ma non piu' i valori negativi  
unsigned long int = vuol dire un numero intero senza segno molto grande

ovviamente se voglio sapere la dimensione occupata in memoria uso

`sizeof(unsigned long int)`

Nel progetto allegato a questa puntata, trovate un programma con cui iniziare a giocare con queste nozioni.

I piu' curiosi, possono implementare il programmino che ho fatto, ad esempio, facendo calcolare la dimensione di tipi di dati composti con i modificatori sopra descritti.

Spero non vi sia venuto il mal di testa...