

Il programmatore che c'e' in noi – Lezione 8– Array di caratteri, stringhe e costanti carattere

Nel corso delle precedenti puntate abbiamo imparato (spero) a conoscere gli array sia di numeri che di caratteri. Nella creazione di programmi si ha spesso a che fare con del testo, il linguaggio C non ha un proprio tipo di dati per gestire il testo, ma usa gli array di caratteri terminati da uno speciale carattere chiamato NUL. (che vale 0 decimale 0x0 esadecimale '\0' costante carattere)

Tanto per essere chiari:

```
char nome[10]; //Dichiara un array di 10 caratteri (indicizzabile tra 0 e 9)
```

```
nome[0]='F';  
nome[1]='r';  
nome[2]='a';  
nome[3]='n';  
nome[4]='c';  
nome[5]='o';  
nome[6]='\0';
```

Ora nome e' una stringa cioe' "Franco".

La prima implicazione e' che quando io devo dichiarare la dimensione del mio array che dovra' contenere una stringa DEVO aggiungere 1 alla dimensione che mi serve, in quanto devo considerare il carattere NUL che devo aggiungere per terminare la stringa.

Nell'esempio indicato il testo massimo che potra' contenere la stringa nome, e' di 9 caratteri + 1 NUL (totale 10 singoli caratteri, come dichiarato all'inizio)

L'istruzione

```
nome[6]='\0';  
puo' essere tranquillamente sostituita con  
nome[6]=0x0; //NUL valore esadecimale  
oppure  
nome[6]=0; //NUL valore decimale
```

Fate attenzione a non scrivere

```
nome[6]='0'; //In questo caso assegno la cifra 0 e non NUL !!! Quindi nome non e' stringa, ma SOLO un array di caratteri.
```

Tutte le lettere e le cifre che utilizziamo, possono essere rappresentate da una costante di tipo carattere oppure dall'equivalente codice ASCII.

Esempio:

Il carattere A si indica con 'A' come costante carattere oppure come valore numerico 65 decimale in codice ASCII.

Tra le costanti carattere segnalo quelle particolari, ossia codici che non hanno un equivalente nella nostra lingua ma hanno un significato speciale se usate nei programmi e che sono rappresentate dal fatto di avere il simbolo BACKSLASH \ seguito da un carattere il tutto racchiuso tra singoli apici.

Tabella delle costanti carattere con backslash

Codice	Significato
'\b'	Backspace
'\f'	Form feed
'\n'	New line

<code>'\r'</code>	Carriage return
<code>'\t'</code>	Horizontal Tab
<code>'\"'</code>	Double quote
<code>'\''</code>	Single Quote
<code>'\\'</code>	Backslash
<code>'\xN'</code>	Hexadecimal constant N
<code>'\N'</code>	Octal constant N

La costante NUL di cui abbiamo parlato in precedenza, quindi altro non e' che `'\0'` ossia zero il ottale (che vale comunque zero in decimale...)

Un ultima considerazione, la costante `'\n'` in realta' e' formata da due caratteri il carattere Carriage return (0xA) o 10 ed il carattere line feed (0xD) o 13, e generalmente indica la fine di una riga in un file di testo (a volte dipende dal sistema operativo...).

Per mettere in pratica quanto esposto ho realizzato il programmino scaricabile qui

Per gestire le stringhe in C language, si utilizzano delle funzioni di libreria, quali `strcpy()` per assegnare un valore, `strcmp()` per confrontare due stringhe e `printf()` per visualizzare il valore di una variabile di tipo stringa. Per informazioni sull'uso delle funzioni di manipolazione delle stringhe, e' sufficiente cercare in internet, troverete innumerevoli esempi e spiegazioni.

Dato che l'obiettivo e' quello di programmare in Cocoa, vi anticipo che tramite Cocoa e quindi Objective C e' comunque possibile gestire le stringhe come array di caratteri ed usare le funzioni standard di libreria, ma Cocoa ha una apposita CLASSE, di nome `NSString` e sue derivate, `NSMutableString` ecc. per gestire le stringhe, di conseguenza ne parleremo quando inizieremo a parlare di Cocoa.

Saluti.