Approfondimento sulle espressioni

Prof. Orazio Mirabella

Espressioni aritmetiche

Si definisce *espressione aritmetica* un insieme di variabili, costanti e richiami di funzione connessi da operatori aritmetici. Il risultato di un'espressione aritmetica è sempre un valore numerico.

negazione (-unario)

moltiplicazione (*), divisione (/), modulo (%)

somma (+), sottrazione (-)

= (assegnamento)

All'interno delle espressioni aritmetiche, la priorità degli operatori segue le regole dell'algebra. Tutte le operazioni di negazione sono eseguite per prime, quindi l'espressione è esaminata nuovamente per eseguire tutte le moltiplicazioni, divisioni e le operazioni modulo. Infine l'espressione viene sottoposta a scansione ancora una volta per eseguire le addizioni e le sottrazioni.

Espressioni logiche

- Un'*espressione logica* è un'espressione che genera come risultato un valore vero o falso (in C non esiste il tipo booleano, presente in altri linguaggi), e viene utilizzata dalle istruzioni di controllo.
- Le espressioni logiche, producono come risultato **1** per vero e **0** per falso (qualsiasi valore numerico diverso da zero viene comunque considerato vero)

> (maggiore di)	>= (maggiore uguale)
< (minore di)	<= minore uguale)
== (uguaglianza)	!= (disuguaglianza)

L'operatore di uguaglianza **==** è diverso, anche nella notazione, da quello di assegnamento **=**

La priorità di **>**, **>=**, **<**, e **<=** è la stessa ed è maggiore di **==** e di **!=**.

Espressioni logiche

- Gli *operatori logici* consentono invece di concatenare fra di loro più espressioni logiche e di negare il risultato di un'espressione logica.
- Esistono tre operatori logici

! (NOT logico)

&& (AND logico)

|| (OR logico)

x	y	x&&y	x||y	!x
0	0	0	0	1
0	1	0	1	1
1	0	0	1	0
1	1	1	1	0

Tavola della verità

Un'espressione può contenere una combinazione di operatori aritmetici, logici e relazionali. L'espressione
Es.: x + sereno + (i<100)

Espressioni logiche

- Uso del connettivo && (AND)
  X==Y && a>b risulta vera se ?
- Uso del connettivo || (OR)
  b<c || t!=r restituisce falso solo se ?
- Uso del connettivo! (Not)
  !y restituisce falso se ?
- Un'espressione può contenere una combinazione di operatori aritmetici, logici e relazionali. L'espressione

  x + sereno + (i<100)

  restituisce la somma di x, di sereno e del risultato di i<100, che è 1 se i è minore di 100, zero altrimenti

Espressioni condizionali

- Una *espressione condizionale* si ottiene con l'operatore ternario ?:

  espr1 ? *espr2* : *espr3*

  se *espr1* è vera, il risultato è *espr2*, altrimenti è *espr3*.

x==y ? a : b significa: "se x è uguale a y, allora a, altrimenti b"

Espressioni condizionali

- Si può utilizzare l'operatore ?: per assegnare un valore a una variabile

  v= x==y ? a*c+5 : b-d;

  Se x è uguale a y, a v viene assegnato il valore di a*c+5, altrimenti gli viene assegnato il valore di b-d.

- *espr1*, *espr2*, *espr3* sono valutate prima di ?:, l'assegnamento per ultimo

- L'espressione condizionale può essere sempre sostituita da un if

```
if(x==y)  v = a*c+5;
else      v = b-d;
```

Variabili carattere (char)

- Le variabili di tipo carattere assumono valori alfanumerici che comprendono le lettere dell'alfabeto minuscole e maiuscole, le cifre decimali, la punteggiatura e altri simboli.

- Scrivendo: char x, y, z;

  la parola chiave char specifica che gli identificatori x, y e z che la seguono si riferiscono a variabili di tipo carattere.

- La definizione fa sì che venga riservato uno spazio in memoria sufficiente per contenere un carattere alfanumerico.

- La dimensione può variare rispetto all'implementazione; molte versioni riservano per i char uno spazio di un byte

Variabili carattere (char)

- Per assegnare un valore costante a una variabile char lo si deve racchiudere tra apici singoli:
- x = 'A'; y = ';'; z = '&';

- A ogni carattere presente nel codice corrisponde una rappresentazione numerica univoca, per cui è possibile confrontare due simboli per mezzo di tutti gli operatori relazionali.
- Es: “A” (65) è maggiore di “;” (59) che a sua volta è maggiore di “&” (38).
- Es:
- if(x=='A')
  printf("Si tratta di una A");
  Oppure
- if(x>='A' && x<='Z')
  printf("Si tratta di una lettera maiuscola");

Stampa di Variabili carattere

- Per poter visualizzare dei char con una printf si deve come al solito indicarne il formato; per esempio:
  printf("%c %c %c", x, y, z);

- i simboli di percentuale tra i doppi apici definiscono il formato di stampa delle corrispondenti variabili; le c (*character*) specificano che si tratta di caratteri.
- Es:
- char x, y, z;
- ……
- x = 'A';
- y = ';';
- z = '&';
- printf("%c %c %c", x, y, z);

- L’esecuzione dell’istruzione restituisce
  A ; &