

Programmazione Procedurale

Elementi di Base

Istruzioni di Assegnazione

versione 3.0

Questo lavoro è concesso in uso secondo i termini di una licenza Creative Commons
(vedi ultima pagina)

G. Mecca Università della Basilicata giansalvatore.mecca@unibas.it



Elementi di Base: Assegnazioni >> Sommario

Sommario

- Introduzione
- Istruzioni di Assegnazione in C++
- Compatibilità tra i Tipi

G. Mecca - Programmazione Procedurale

2

Introduzione

- Assegnare un valore ad una variabile
 - ❑ primo modo: istruzione di assegnazione
 - ❑ secondo modo: lettura del valore da un dispositivo esterno (es: tastiera o file)
- Istruzioni di assegnazione
 - ❑ funzionalità essenziale di ogni linguaggio
 - ❑ servono ad effettuare calcoli e a conservare i risultati nelle variabili

Istruzione di Assegnazione in C++

- Due elementi principali

- ❑ identificatore della variabile (parte sinistra)
- ❑ espressione (parte destra)

ATTENZIONE: senza float
(il tipo si indica solo nella
dichiarazione della variabile)

- Esempio:

cerchio = raggio * raggio * pigreco;

- Intuitivamente:

- ❑ calcola il valore dell'espressione e lo assegna alla variabile

Istruzione di Assegnazione

○ Sintassi

`<nomeVariabile> = <espressione>;`

○ Dove

- ☐ `<nomeVariabile>` è un identificatore
- ☐ `=` è l'operatore di assegnazione
- ☐ `<espressione>` è una formula per il calcolo di un valore (`>>`)

Istruzione di Assegnazione

○ Semantica

- ☐ attenzione: il simbolo di uguale non deve essere inteso come un simbolo di equazione
- ☐ l'istruzione ha una precisa semantica

○ Semantica "operazionale"

- ☐ descrive le operazioni effettuate dal processore per eseguire l'istruzione
- ☐ per interpretare le istruzioni è necessario attenersi alla semantica operazionale

Istruzione di Assegnazione

- Semantica “operazionale” dell’assegnazione
 - ☐ viene calcolato il valore dell’espressione a partire dai valori delle costanti e delle variabili
 - ☐ se il valore risultante è compatibile con il tipo della variabile, viene assegnato alla variabile (ovvero: il valore viene scritto nello spazio di memoria associato alla variabile)
 - ☐ altrimenti si genera un errore

Istruzioni di Assegnazione

- Calcolo del valore dell’espressione
 - ☐ il processore applica gli operatori agli operandi
 - ☐ prelevando gli opportuni valori dalla memoria (costanti simboliche, variabili)
- Nota sul calcolo dell’espressione
 - ☐ il calcolo viene effettuato utilizzando registri di lavoro interni al processore
 - ☐ e non modifica il valore degli operandi

Istruzione di Assegnazione

○ Esempi:

```
int a, b, c;
```

```
b = 1;
```

```
a = b * 3;
```

```
a = abs(-2) + 5;
```

```
a = a + 1;
```

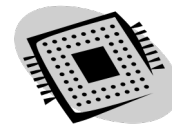
```
b = c;
```

```
c = 4;
```

Memoria RAM

...		
#1000	a	8
#1001	b	56321
#1002	c	56321
#1003		
#1004		
...		

Processore



R1 8

R2 2

ATTENZIONE: dopo aver cambiato il valore di b con l'istruzione b = c il valore di a NON cambia (NON si tratta di equazioni matematiche)

Compatibilità tra i Tipi

○ Caso a

- ☐ il tipo del risultato dell'espressione è lo stesso della variabile

☐ es: **int a, b;**

```
b = 10;
```

```
a = b * 12 - 34;
```

☐ es: **float a, b;**

```
b = 10.1;
```

```
a = b * 12.0 / 34;
```

Compatibilità tra i Tipi

- Caso b

- ☐ il tipo del risultato dell'espressione è diverso dal tipo della variabile
- ☐ i due tipi sono tipi numerici (interi o reali)

- In questo caso

- ☐ avviene una conversione implicita del valore dell'espressione
- ☐ l'assegnazione viene effettuata comunque

Compatibilità tra i Tipi

- Il valore dell'espressione è convertito nel tipo della variabile

- Due possibili tipi di conversione

- ☐ Ampliamento: conversione da un tipo meno generale ad un tipo più generale
es: valore intero per variabile float
- ☐ Troncamento: caso contrario al precedente
es: valore float per variabile intera

Compatibilità tra i Tipi

ATTENZIONE
il troncamento
produce una perdita
di valore

○ Ampliamento

```
int c; float d;
c = 10;      // il valore di c è 10
d = 10;      // il valore di d è 0.1E+2
```

○ Troncamento

```
int a;
a = .12e+4;  // il valore di a è 1200
a = 2.32;    // il valore di a è 2
```

Riassumendo

○ Istruzioni di assegnazione

- ☐ <variabile> = <espressione>
- ☐ semantica “operazionale” (ATTENZIONE)
- ☐ compatibilità tra tipi (ATTENZIONE)

Termini della Licenza

Termini della Licenza

- This work is licensed under the Creative Commons Attribution-ShareAlike License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-sa/1.0/> or send a letter to Creative Commons, 559 Nathan Abbott Way, Stanford, California 94305, USA.
- Questo lavoro viene concesso in uso secondo i termini della licenza “Attribution-ShareAlike” di Creative Commons. Per ottenere una copia della licenza, è possibile visitare <http://creativecommons.org/licenses/by-sa/1.0/> oppure inviare una lettera all’indirizzo Creative Commons, 559 Nathan Abbott Way, Stanford, California 94305, USA.