

- La conversione fra interi di differenti dimensioni deve essere il più possibile evitata. La conversione di un intero di grandi dimensioni a uno più piccolo (da 32 a 16 bit o da 16 a 8 bit) può causare il troncamento del valore memorizzato in una variabile o determinarne il cambio di segno. Ad esempio convertire l'intero signed a 16 bit -1 in intero unsigned a 32 bit darà come risultato il valore 4.294.967.295

In particolare sono negate tutte le conversioni riportate nella seguente tabella:

Da	A
16 bit signed	32 bit unsigned
32 bit signed	16 bit unsigned
32 bit unsigned	16 bit signed
32 bit signed	16 bit signed

Il codice non deve affidarsi a conversioni implicite e/o dedotte dal compilatore.

7.1.8.10 Computazione e condizionali

- I dati devono essere appropriatamente confrontati con altri dello stesso tipo, specialmente per i tipi float e double.

Esempio:

if (variabile == 0.1) questa condizione potrebbe non rivelarsi mai vera, per le proprietà di arrotondamento del compilatore;

- Le variabili dichiarate come unsigned non devono mai essere confrontate con lo zero utilizzando l'operatore "maggiore di".

Esempio: if (variabile > 0) risulta sempre vero se variabile è unsigned;

- Le variabili dichiarate come signed, non devono mai essere confrontate con TRUE.

Esempio: if (variabile)

Se ad esempio variabile può assumere un valore negativo è meglio prevedere questo caso con un controllo del tipo: if (variabile != 0) oppure ancora più esplicito controllando il segno dell'intero.

7.1.8.11 Controllo del flusso

Variabili di controllo

È obbligatorio utilizzare sempre un limite superiore "inclusive" e il limite inferiore come "esclusive".

Esempio:

Forma non corretta:

`x >= 23 e x <= 42`

Forma corretta:

`x >= 23 e x < 43`

Switches

- Ogni blocco di codice appartenente a ogni "case" di uno switch deve essere terminato dalla keyword "break";
- Ogni switch deve avere un "case" di default.

7.1.8.12 Passaggio di argomenti

- I tipi di dati esterni non devono essere passati "per valore" (by value);
- I vettori e le strutture devono sempre essere passati per indirizzo o per riferimento;
- È auspicabile utilizzare la keyword "const" per i parametri costanti (strutture o vettori) passati in ingresso a una funzione.

7.1.8.13 Valori di ritorno

I tipi di dati devono essere appropriati per memorizzare i valori di ritorno delle funzioni;