

## 4.3 Inizializzazione di variabili

L'inizializzazione di una variabile può essere esplicitata direttamente al momento della sua dichiarazione, come con

```
int i = 0;
```

L'istruzione dichiara la variabile `i` di tipo intero e le assegna il valore zero. Di seguito alla dichiarazione di tipo possono essere definite e inizializzate più variabili:

```
int a = 50, b = 30, c, d = 333;
```

definisce le variabili intere `a`, `b`, `c` e `d`; inizializza `a` al valore 50, `b` a 30, `d` a 333, `c` non è inizializzata. Analogamente si possono assegnare valori agli altri tipi di variabili semplici:

```
float x = 567.8927;  
float y = 7e13;  
char risposta = 's';
```

Per gli array l'inizializzazione è possibile solamente se sono stati dichiarati come `extern` o come `static`; quest'ultima classe di variabile verrà esaminata in seguito.

Le variabili `extern` sono quelle che vengono definite prima di `main`. L'inizializzazione si ottiene inserendo i valori tra parentesi graffe, separati da una virgola:

```
int voti[6] = {11, 18, 7, 15, 21, 9};
```

Il compilatore fa la scansione dei valori presenti tra parentesi graffe da sinistra verso destra e genera altrettanti assegnamenti consecutivi agli elementi del vettore, rispettando la loro posizione; dunque `voti[0]` assume il valore 11, `voti[1]` 18, `voti[2]` 7 ecc. Quando tutti gli elementi dell'array vengono inizializzati è possibile omettere l'indicazione del numero di elementi, e scrivere

```
int voti[] = {11, 18, 7, 15, 21, 9};
```

È infatti il compilatore stesso che conta i valori e di conseguenza determina la dimensione del vettore.

Gli array di caratteri, comunemente detti *stringhe*, possono essere inizializzati anche inserendo il loro contenuto tra doppi apici:

```
char frase[] = "Analisi, requisiti";
```