

12.4 Campi

I *campi* sono un costrutto del C che consente di far riferimento in modo simbolico ai singoli bit di una variabile:

```
struct {
    unsigned k: 1;
    unsigned j: 6;
} var_bit_bit;
```

La variabile `var_bit_bit` occupa uno spazio di memoria pari alla somma dei bit utilizzati da ogni campo arrotondata alla *word* (parola) della macchina. In altri termini, supponendo che la parola della macchina sia di 16 bit, si ha che:

- • il campo `k` di `var_bit_bit` occupa 1 bit;
- • il campo `j` di `var_bit_bit` occupa 6 bit;
- • 9 bit rimangono inutilizzati.

I campi possono essere trattati alla stregua dei membri di una struttura e quindi essere utilizzati in espressioni come la seguente

```
var_bit_bit.k = 1;
```

usata per esempio per alzare a uno il flag `k` della variabile `var_bit_bit`. La sintassi generale dei campi è:

```
struct {
    tipo_campo nome_campo1 : numero_bit_1;
    tipo_campo nome_campo2 : numero_bit_2;
    ...
    tipo_campo nome_campoN : numero_bit_N;
} var_1, var_2, var_N;
```

oppure equivalentemente:

```
struct nome_struct {
    tipo_campo nome_campo1 : numero_bit_1;
```

```
    tipo_campo nome_campo2 : numero_bit_2;  
    ...  
    tipo_campo nome_campoN : numero_bit_N;  
};  
struct nome_struct var_1, var_2, var_N;
```

I campi non hanno incontrato un grande successo presso i programmatori C, poiché di fatto non producono ottimizzazione né di tempo né di spazio, cosicché al loro posto si preferiscono le classiche operazioni bit a bit.