



Laboratorio di Programmazione 1

**Docente: dr Damiano Macedonio
Lezione 14 - 30/01/2014**

1

Funzioni:

Variabili Locali e Globali

- Le variabili definite all'interno di una funzione sono dette *locali*.
 - Il valore di una variabile locale può essere utilizzato solo all'interno della funzione in cui la variabile è definita.
- Le variabili definite all'esterno di qualsiasi funzione dette *globali*.
 - Il valore di una variabile globale può essere utilizzato da qualsiasi funzione del programma: qualsiasi funzione può modificarne il valore.
 - Le variabili globali sono definite per prime nel programma.

Funzioni: Variabili Locali e Globali

```
#include <stdio.h>
int multiplier;          // variabile globale inizializzata a 0

int mult( int a ){
    return a * multiplier;    // uso della variabile globale
}

void changeMultiplier( int m ){
    multiplier = multiplier + m;    // modifica variabile globale
}

int main( void ){
    int a = 5;
    changeMultiplier( 10 );
    printf( "%i * %i = %i", a, multiplier, mul( a ) ); // 5*10
    changeMultiplier( 8 );
    printf( "%i * %i = %i", a, multiplier, mul( a ) ); // 5*18
    return 0;
}
```

Funzioni:

Variabili Locali e Globali

- Le variabili globali possono essere utilizzate nei programmi in cui più funzioni devono accedere al valore di una stessa variabile.
 - Permette di evitare il passaggio esplicito di un parametro.
 - Limita la leggibilità del programma.
 - Riduce la genericità della funzione.
- Valore di default:
 - Il valore di default di una *variabile globale* è zero
 - `int a;` // la variabile `a` è inizializzata a 0
 - `int array[10]` // tutti gli elementi sono inizializzati a zero.
 - Le *variabili locali* non hanno un valore iniziale di default e quindi devono essere esplicitamente inizializzate nel programma.

Funzioni:

Variabili Automatiche e Statiche

- Le variabili locali dichiarate all'interno di una funzione sono dette *automatiche*:
 - Una variabile automatica è creata ogni volta che la funzione viene creata.
 - Quando la funzione termina, queste variabili “scompaiono”.
- Le variabili locali possono essere dichiarate statiche, usando la parola chiave `static` davanti alla dichiarazione della variabile.
 - Una variabile locale statica è creata una sola volta quando viene avviato il programma principale.
 - Quando la funzione termina, la variabile mantiene il suo valore per una prossima invocazione della funzione.
 - Le variabili locali statiche hanno un valore di `default` pari a zero.

Funzioni: Variabili Automatiche e Statiche

```
#include <stdio.h>

void auto_static( void ){
    int a = 1;
    static int b = 1;           // variabile statica
    printf( "Variabile automatica: %i, variabile statica: %i", a, b );
    a = a+1;
    b = b+1;
}

int main( void ){
    int i;
    for( i = 0 ; i < 10; i++ ){
        auto_static(); // a sempre 1, b valore da 1 a 10
    }
    return 0;
}
```

Funzioni:

Variabili Automatiche e Statiche

- Le variabili statiche sono usate per mantenere il valore di una variabile tra diverse invocazioni della stessa funzione.
- Le variabili statiche sono particolarmente utili quando una funzione usa una variabile il cui valore viene impostato una sola volta e non cambia più.
 - Si evita l'inefficienza di dover reinizializzare la variabile ogni volta. Questo è particolarmente utile con gli array.

Funzioni Ricorsive

- Una funzione è detta *ricorsiva* se include una chiamata a sé stessa.
- Le funzioni ricorsive sono utili quando la soluzione di un problema può essere ottenuta applicando successivamente la stessa soluzione a sotto-problemi più piccoli.

Funzioni Ricorsive

```
#include <stdio.h>

int factorial( int n ){
    int result;
    if( n == 0 ){ // caso base
        result = 1;
    } else {      // passo ricorsivo
        result = n * factorial( n - 1 );
    }
    return result;
}
```

```
int main( void ){
    int i;
    for( i = 0; i < 11; i++ ){
        printf( "Il fattoriale di %i e' %i.\n", i, factorial(i) );
    }
    return 0;
}
```

Anche se viene chiamata la stessa funzione, è considerata come una chiamata ad una funzione distinta: la funzione acquisisce il suo insieme di parametri formali e variabili locali.