

2 - DICHIARAZIONE E USO DELLE VARIABILI

Gli elementi di base di un programma sono le variabili e le costanti.

In C le variabili devono essere dichiarate prima di essere usate. Le variabili fondamentali che si possono definire sono di tipo: char, int, float, double e void.

Ad esempio con char si definiscono variabili di tipo carattere, con int variabili che possono contenere solo numeri interi, con float e double variabili che possono contenere numeri reali. Il tipo void per ora non verrà utilizzato.

Per dichiarare una variabile basta scrivere il tipo seguito dal nome che decidiamo di associare alla variabile stessa, ad esempio:

```
int c1;
```

In questo caso abbiamo dichiarato una variabile di tipo intero e di nome c1. Si noti che abbiamo terminato la dichiarazione con un ";". Il nome da dare alla variabile si chiama identificatore può essere scelto a piacere, rispettando alcune limitazioni.

Le parole chiave come printf() devono essere scritte in minuscolo, e solitamente anche le variabili sono scritte con caratteri minuscoli, nonostante si possano usare anche caratteri maiuscoli.

La regola quindi sarebbe di scrivere le variabili a lettere minuscole, e se si vuole si può usare il carattere "_" come ad esempio:

```
int c1_prova;
```

Detto questo, vediamo come assegnare un valore ad una variabile:

```
c1 = 1000;
```

Abbiamo introdotto l'**operatore di assegnamento "="**, che in questo caso assegna a c1 il valore 1000.

Si noti che:

```
c1 = 500+500;  
c1 = (250+250)+500;
```

sono istruzioni equivalenti, infatti si può assegnare anche un'espressione.

Per stampare il valore di una variabile sul video con la funzione printf() occorre aggiungere %d nel punto che ci interessa, e mettere il nome della variabile dopo la chiusura delle "", in questo modo:

```
printf("Il valore di c1 e' %d.", c1);
```

Al momento della stampa il %d sarà sostituito dal valore della variabile c1. Da notare la virgola che separa i doppi apici "" dalla variabile.

Esempio:

```
#include <stdio.h>          /* Includiamo la libreria standard */
int c1;                    /* Dichiaro c1, una variabile di tipo intero */
int main()                 /* Funzione principale, eseguita per prima */
{                           /* Inizio della funzione main() */
    c1 = 1000;              /* Assegno a c1 il valore 1000 */
    printf("Il valore di C1 e' %d.\n",c1);
    c1 = 1500;             /* Assegno a c1 il valore 1500 */
    printf("Il valore di C1 e' %d.\n",c1);
    return(0);
}                           /* Fine della funzione main() */
```

Come risultato si ottiene:

Il valore di C1 e' 1000.

Il valore di C1 e' 1500.

Si noti che abbiamo usato %d per indicare che la variabile in questione è di tipo intero. Per stampare il valore di una variabile di tipo float, dovremmo mettere %f. Vedremo in seguito altre opzioni di stampa.

Si noti che il C è un linguaggio case-sensitive, pertanto se scrivessimo Printf il compilatore darebbe un errore in quanto non troverebbe la funzione nella libreria stdio.h.

Per dichiarare due variabili dello stesso tipo conviene scrivere:

```
int c1, c2;
```

anziché dichiarare singolarmente.

Per stampare più di una variabile numerica occorre mettere un %d o %f nel punto del testo che ci interessa, e mettere poi il nome della variabile dopo la stringa.

Esempio:

```
printf("C1 = %d , C2 = %d\n", c1, c2);
```

Ossia si mettono i %d o %f dove interessa, e poi la lista delle variabili che li sostituiranno nello stesso ordine, separate da virgole.

Ed ora proviamo quanto imparato fino ad ora:

```
#include <stdio.h>
#include <conio.h>
int c1,c2;
int main()
{
    c1 = 1000;
    c2 = 1500;
    printf("C1 = %d , C2 = %d\n", c1, c2);
    c1 = 200;
    c2 = c1*7+3;
    printf("C1 = %d , C2 = %d\n", c1, c2);
}
```

```
printf("Premi un tasto per terminare\n");
getch();
return(0);
}
```

ESERCITAZIONI:

Scrivere un programma che visualizza il contenuto di una variabile prima e dopo averla incrementata di 3.

Soluzione:

```
#include <stdio.h>
#include <conio.h>
int x;
int main()
{
    x = 1;
    printf("x = %d \n", x);
    x = x + 3;
    printf("x = %d \n", x);
    printf("Premi un tasto per terminare\n");
    getch();
    return(0);
}
```

Scrivere un programma che calcola e visualizza il risultato della seguente espressione:

$$(37-(3*(2+1)))/7$$

Soluzione:

```
#include <stdio.h>
#include <conio.h>
int r;
int main()
{
    r = (37-(3*(2+1)))/7;
    printf("Risultato = %d \n", r);
    printf("Premi un tasto per terminare\n");
    getch();
    return(0);
}
```