

# Esercitazione Programmazione Procedurale con Laboratorio

## Comandi Condizionali e Cicli

### Esercizio 1

Si scriva un programma in linguaggio C che legga due numeri da tastiera, con identificatore  $a$  e  $b$ , e determini le seguenti informazioni, stampandole a video:

- determini se  $b$  è un numero positivo o negativo;
- determini se  $a$  è un numero pari o dispari;
- calcoli il valore di  $a + b$ ;
- determini il valore massimo della somma di  $a$  e  $b$  senza considerare il loro segno (utilizzare il loro valore assoluto).

### Soluzione dell'esercizio 1

```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 int main(void) {
5
6     int a= 0, b= 0;
7     int c= 0;
8
9     printf("Immetti A: ") ;
10    scanf("%d", &a) ;
11    printf("Immetti B: ") ;
12    scanf("%d", &b) ;
13
14    if( b >= 0 )
15        printf("B e' positivo\n");
16    else
17        printf("B e' negativo\n");
18
19    if( a%2 == 0 )
20        printf("A e' pari\n") ;
21    else
22        printf("A e' dispari\n") ;
23
24
25    c=a+b;
26    printf("La somma %d + %d e' uguale a %d\n", a, b, c) ;
27
28    if( a < 0 )
29        a = -a ;
30    if( b < 0 )
31        b = -b ;
32
33    printf("Il valore massimo della somma +-A +-B e' uguale a %d\n", a+b );
34    return 0;
35 }
```

### Esercizio 2 Cosa stampa?

Dato il seguente programma (senza implementarlo), scrivere cosa stampa.

```

1 #include <stdio.h>
2
3 int main(void) {
4
5     int a= 6,
6     b= 7, c= 13;
7
8     do {if (a<b) if(b%2 !=0) if(4) b++; if(c == 2*a) c++; else c--; if (a+b>=c?1:0) a--;} while(a>0);
9
10    printf("a= %d", a);
11    printf("b= %d", b);
12    printf("c= %d", c);
13
14 }

```

## Soluzione dell'esercizio 2

a= 0, b= 8, c= 7

## Esercizio 3

Leggere un carattere (*char*) da tastiera e stampare “Maschio” se “M” o “m”, “Femmina” se “F” o “f”, oppure “Sesso non specificato” se diverso da questi caratteri. Utilizzare il comando *switch*.

## Soluzione dell'esercizio 3

```

1 #include <stdio.h>
2
3 int main() {
4     char gender;
5
6     printf("Enter gender (M/m or F/f): ");
7     scanf("%c",&gender);
8
9     switch(gender) {
10        case 'M':
11        case 'm':
12            printf("Male.");
13            break;
14        case 'F':
15        case 'f':
16            printf("Female.");
17            break;
18        default:
19            printf("Unspecified Gender.");
20    }
21
22    printf("\n");
23    return 0;
24 }

```

## Esercizio 4 While ed espressioni di controllo

Rispondere alle domande nei commenti:

```

1 // 1) Che funzione implementa questo programma?
2 // 2) Cosa succede se da tastiera passate un numero minore di 0?
3 #include <stdio.h>
4
5 int main() {
6     int n= 0, res= 1;
7     scanf("%d", &n);
8
9     while (n-->0)
10        res*= (n + 1);
11
12    printf("%d\n", res);
13
14    return 0;
15 }

```

## Soluzione dell'esercizio 4

Il primo programma calcola il fattoriale di  $n$  (passato da tastiera). Se  $n$  è negativo, il ciclo termina non correttamente, dato che qualsiasi valore diverso da 0 in C equivale a *true*. Più precisamente (sulla mia piattaforma),  $n$  viene decrementato fino al minimo valore rappresentabile con un tipo *int* (se *int* quattro byte,  $-2,147,483,648$ , rappresentato con 32 bit con valore 1), il successivo decremento porta  $n$  al massimo valore rappresentabile con un *int* (se *int* quattro byte,  $2,147,483,647$ ); situazione di *overflow*. Infine  $n$  viene decrementato di volta in volta fino a terminare quando  $n = 0$ .

## Esercizio 5      Loops

- Utilizzare un ciclo *while*, un *do...while*, e un *for* per stampare tutti i numeri tra 0 e  $n$  con il valore  $n$  letto da tastiera (*scanf()*). Per esempio, se  $n = 10$ , la stampa deve essere 12345678910.
- Calcolare il massimo comune divisore (MCD) tra due valori di tipo *int* presi in input da tastiera. Per farlo, scandire tutti i valori tra 1 ed il minimo dei due valori, controllando per ogni valore scandito se è divisore di entrambi. Il massimo di essi corrisponde al MCD.

## Soluzione dell'esercizio 5

Di seguito, la soluzione solo per il secondo punto.

```
1 #include <stdio.h>
2
3 int main(void) {
4     int i, num1, num2, min, gcd=1;
5
6     printf("Ricevo i due numeri per i quali calcolare il GCD: ");
7     scanf("%d %d", &num1, &num2);
8
9     min = (num1 < num2) ? num1 : num2;
10
11    for(i=1; i <= min; i++) {
12
13        // Se i e' fattore di entrambi i numeri
14        if(num1 % i == 0 && num2 % i == 0)
15            gcd = i;
16    }
17
18    printf("GCD di %d e %d = %d\n", num1, num2, gcd);
19    return 0;
20 }
```

## Esercizio 6      Uso di *break* e *continue*

Modificare il seguente programma in modo che: i) se si legge da tastiera un valore minore di 0 si esce dal ciclo, ii) se si legge un valore maggiore di 10, la lettura da tastiera viene ripetuta ancora. Il punto ii) può essere ripetuto 5 volte: alla quinta volta, se il valore è sempre maggiore di 10, si esce dal ciclo. Utilizzare i comandi *break* e *continue*.

```
1 #include <stdio.h>
2
3 int main(void) {
4     int a= 0;
5
6     while (1) {
7         scanf("%d", &a);
8
9         // Aggiungere comandi qui
10
11        a++;
12    }
13
14    printf("Valore di a= %d", a);
15    return 0;
16 }
```

## Soluzione dell'esercizio 6

```

1 #include <stdio.h>
2
3 int main(void) {
4     int a= 0;
5     int counter= 0;
6
7     while (1) {
8         scanf("%d", &a);
9         if (a < 0) {
10            puts("Errore, lettura minore di 0\n");
11            break;
12        }
13        else
14            if (a > 10) {
15                counter++;
16                if (counter >= 5) {
17                    puts("Errore, lettura cinque volte maggiore di 10\n");
18                    break;
19                }
20                else
21                    continue;
22            }
23        a++;
24    }
25    return 0;
26 }

```

## Esercizio 7 Disegno figure geometriche

1. Si realizzi un programma in linguaggio C che legga un numero intero N e visualizzi un quadrato di asterischi di lato N (vedi esempio con N = 5).
2. Si realizzi una variante del programma per visualizzare solo i lati del quadrato (vedi esempio con N = 5).
3. Si realizzi una variante del programma per visualizzare un triangolo isoscele rettangolo di lato N (vedi esempio con N = 5).
4. Si realizzi una variante del programma per visualizzare un quadrato di lato N come nell'esempio del caso 4 (con N = 5).

Caso 1	Caso 2	Caso 3	Caso 4
*****	*****	*	*++++
*****	* * *	**	*++++
*****	* * *	***	*+++*
*****	* * *	****	*****
*****	*****	*****	******

### Soluzione dell'esercizio 7

```

1 #include <stdio.h>
2 #include <stdlib.h>
3
4 int main(void) {
5     int lato; /* lato del quadrato */
6
7     int riga, colonna;
8
9     /* riga e colonna del quadrato */
10    /* leggi il lato del quadrato */
11
12    printf("inserisci il lato del quadrato: ");
13    scanf("%d", &lato);
14

```

```

15 /* controlla se il lato del quadrato e' un numero maggiore di 0 */
16 if ( lato <= 0 )
17     printf("errore, il lato deve essere maggiore di zero\n");
18 else {
19
20     /* il ciclo piu' esterno scandisce la righe del quadrato */
21     /* inizializza la variabile per la scansione delle righe del quadrato */
22     riga = 0;
23     while ( riga < lato ) {
24         /* il ciclo piu' interno scandisce le colonne del quadrato */
25         /* per ogni riga stampa "*" per ogni colonna */
26         /*inizializza la variabile per la scansione delle colonne del quadrato */
27         colonna = 0;
28         while ( colonna < lato ) {
29             /* stampa "*" senza andare a capo */
30             printf("*");
31
32             /* incrementa "colonna" per passare alla colonna successiva */
33             colonna = colonna + 1;
34         }
35
36         /* terminata la stampa di una riga si deve riportare il cursore al margine */
37         /* sinistro dello schermo */
38         printf("\n");
39
40         /* incrementa "riga" per passare alla riga successiva */
41         riga = riga + 1;
42     }
43 }
44 exit(0) ;
45 }
46

```

```

1 #include <stdio.h>
2 #include <stdlib.h>
3
4 int main(void) {
5
6     int lato; /* lato del quadrato */
7     int riga, colonna; /* riga e colonna del quadrato */
8
9
10    /* leggi il lato del quadrato */
11    printf("inserisci il lato del quadrato: ");
12    scanf("%d",&lato);
13
14    /* controlla se il lato del quadrato e' un numero maggiore di 0 */
15    if ( lato <= 0 )
16        printf("errore, il lato deve essere maggiore di zero\n");
17
18    else {
19        /* il ciclo piu' esterno scandisce la righe del quadrato */
20        /* inizializza la variabile per la scansione delle righe del quadrato */
21        riga = 0;
22        while ( riga < lato ) {
23            /* il ciclo piu' interno scandisce le colonne del quadrato */
24            /*inizializza la variabile per la scansione delle colonne del quadrato */
25            colonna = 0;
26
27            while ( colonna < lato ) {
28                /* per la prima e l'ultima riga stampa "*" per ogni colonna */
29                if ( riga == 0 || riga == (lato-1) )
30                    printf("*");
31
32                else {
33                    /* per le altre righe stampa "*" solo per la prima e l'ultima colonna */
34                    if ( colonna == 0 || colonna == (lato-1) )
35                        printf("*");
36                    else
37                        /* in tutti gli altri casi stampa uno spazio*/
38                        printf(" ");

```

```

39     }
40
41     /* incrementa "colonna" per passare alla colonna successiva */
42     colonna = colonna + 1;
43 }
44
45 /* terminata la stampa di una riga si deve riportare il cursore al margine sinistro dello schermo
46 */
47 printf("\n");
48
49 /* incrementa "riga" per passare alla riga successiva */
50 riga = riga + 1;
51 }
52 }
53 exit(0);
54 }

```

```

1 #include <stdio.h>
2 #include <stdlib.h>
3
4 int main(void) {
5
6     int lato; /* lato del triangolo */
7     int riga, colonna; /* riga e colonna del triangolo */
8
9     /* leggi il lato del triangolo */
10    printf("inserisci il lato del triangolo: ");
11    scanf("%d", &lato);
12
13    /* controlla se il lato del triangolo e' un numero maggiore di 0 */
14    if ( lato <=0 )
15        printf("errore, il lato deve essere maggiore di zero\n");
16
17    else {
18        /* il ciclo piu' esterno scandisce la righe del triangolo */
19        /* inizializza la variabile per la scansione delle righe del triangolo */
20        riga = 0;
21
22        while ( riga < lato ) {
23            /* il ciclo piu' interno scandisce le colonne del triangolo */
24            /* per ogni riga stampa "*" solo se colonna <= riga */
25            /* inizializza la variabile per la scansione delle colonne del triangolo */
26            colonna = 0;
27
28            while ( colonna <= riga ) {
29                /* stampa "*" senza andare a capo */
30                printf("*");
31
32                /* incrementa "colonna" per passare alla colonna successiva */
33                colonna = colonna + 1;
34            }
35
36            /* terminata la stampa di una riga si deve riportare il cursore al margine sinistro dello schermo
37            */
38            printf("\n");
39
40            /* incrementa "riga" per passare alla riga successiva */
41            riga = riga + 1;
42        }
43    }
44    exit(0);
45 }
46

```

```

1 #include <stdio.h>
2 #include <stdlib.h>
3
4 int main(void) {
5

```

```

6 int lato; /* lato del quadrato */
7 int riga, colonna; /* riga e colonna del quadrato */
8
9 /* leggi il lato del quadrato */
10 printf("inserisci il lato del quadrato: ");
11 scanf("%d", &lato);
12
13 /* controlla se il lato del quadrato e' un numero maggiore di 0 */
14 if ( lato <= 0 )
15     printf("errore, il lato deve essere maggiore di zero\n");
16 else {
17     /* il ciclo piu' esterno scandisce la righe del quadrato */
18     /* inizializza la variabile per la scansione delle righe del quadrato */
19     riga = 0;
20
21     while ( riga < lato ) {
22         /* il ciclo piu' interno scandisce le colonne del quadrato */
23         /* inizializza la variabile per la scansione delle colonne del quadrato */
24         colonna = 0;
25
26         while ( colonna < lato ) {
27             /* se colonna <= riga stampa "*" altrimenti stampa "+" */
28             if ( colonna <= riga )
29                 printf("*");
30             else
31                 printf("+");
32
33             /* incrementa "colonna" per passare alla colonna successiva */
34             colonna = colonna + 1;
35         }
36
37         /* terminata la stampa di una riga si deve riportare il cursore al margine sinistro */
38         /* dello schermo */
39         printf("\n");
40
41         /* incrementa "riga" per passare alla riga successiva */
42         riga = riga + 1;
43     }
44 }
45 exit(0);
46 }

```