

Controlli Condizionali: If-Else

[Descrizione](#)

[Sintassi](#)

[Funzionamento](#)

[In sintesi](#)

[Esercizi](#)

[ESERCIZIO 1:](#)

[ESERCIZIO 2:](#)

[ESERCIZIO 3:](#)

Descrizione

Un'istruzione if-else controlla il ramo condizionale. Le istruzioni nell'oggetto vengono eseguite solo se l'oggetto `if-branchcondition` restituisce un valore diverso da zero (o `true`). Se il valore di `condition` è diverso da zero, l'istruzione seguente viene eseguita e l'istruzione che segue viene ignorata `else`. In caso contrario, l'istruzione seguente viene ignorata e, se è presente un'istruzione `else` che segue, `else` viene eseguita.

Sintassi

The syntax of the `if` statement is:

LA sintassi dell' `if` statement è:

```
if (condizione) {  
    // corpo dell'if statement  
}
```

L' `if` statement valuta la `condizione` dentro alle parentesi `()`.

- Se la `condizione` risulta `true`, il codice dentro al corpo dell' `if` viene *eseguito*.
- Se la `condizione` risulta `false`, il codice dentro al corpo dell' `if` viene *evitato*.

Nota: Il codice dentro `{ }` è il corpo dell' `if` statement.

Funzionamento

L'istruzione **if** permette di verificare determinate condizioni ed ha la seguente sintassi:

```
//...  
  
if(condizione){  
    //....  
}  
  
//...
```

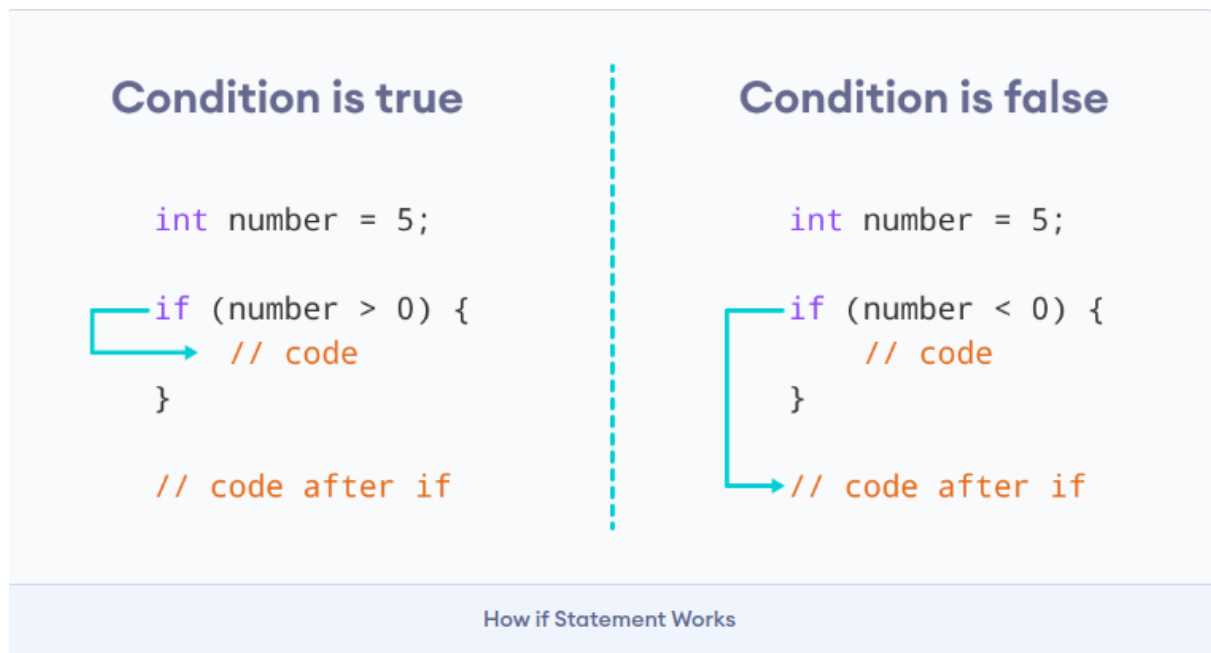
In questo caso **se** l'espressione risulta **vera**, fa eseguire l'istruzione immediatamente successiva, **altrimenti** (se la condizione è **falsa**) si salta l'istruzione (od il blocco di istruzioni) e si procede nell'esecuzione delle istruzioni successive, che possono essere la prosecuzione del programma o un semplice **else**, ad indicare la possibile alternativa all'if:

```
f(condizione){  
    //....  
} else {  
    //...  
}
```

o un **else if** che permette di verificare una o più condizioni:

```
if(condizione1){  
    //....  
}  
else if(condizione2){  
    //...  
} else {  
    //...  
}
```

Così si può comandare il flusso del programma decidendo di eseguire una parte di codice oppure no (nel caso del solo **if**), di fare una scelta tra due parti di codice (nel caso **if– else**) o di fare una scelta tra più parti di codice (nel caso **if – else if– else**).



In sintesi

- Utilizzare l' `if` istruzione per specificare un blocco di codice C++ da eseguire se una condizione è `true`.
- Utilizzare l' `else` istruzione per specificare un blocco di codice da eseguire se la condizione è `false`.
- Utilizzare l' `else if` istruzione per specificare una nuova condizione se la prima condizione è `false`.

esempio:

```
if (risultato_esame >= 18){
    printf ("Complimenti hai superato l'esame");
}
else if (risultato_esame >= 15) {
    printf ("Devi sostenere l'orale per questo esame");
}
else {
    printf ("Non hai superato l'esame");
}
```

Esercizi

ESERCIZIO 1:

realizzare un programma che chiede all'utente 3 valori (valore1, valore2, valore3), in base ai quali il programma restituisce in output i 3 valori dal maggiore al minore.

Esempio Output:

inserisci 3 numeri:

4

3

5

numero3 >= numero1 >= numero2

SOLUZIONE:

```
#include <iostream>
using namespace std;

int main() {
    int numero1;
    int numero2;
    int numero3;

    printf("inserisci 3 numeri: ");
    scanf("%d", numero1);
    scanf("%d", numero2);
    scanf("%d", numero3);

    if (numero1 >= numero2 && numero1 >= numero3) {
        if (numero2 >= numero3) {
            printf("numero1 >= numero2 >= numero3 \n");
        }
        else {
            printf("numero1 >= numero3 >= numero2 \n");
        }
    }
    else if (numero2 >= numero1 && numero2 >= numero3) {
        if (numero1 >= numero3) {
            printf("numero2 >= numero1 >= numero3 \n");
        }
        else {
            printf("numero2 >= numero3 >= numero1 \n");
        }
    }
    else {
        if (numero2 >= numero1) {
            printf("numero3 >= numero2 >= numero1 \n");
        }
        else {
            printf("numero3 >= numero1 >= numero2 \n");
        }
    }
}
```

```
    }  
}  
  
return 0;  
}
```

ESERCIZIO 2:

realizzare un programma che chiede all'utente 2 valori, in base ai quali il programma dice qual è il maggiore, oppure uguali nel caso.

Esempio Output:

inserisci primo valore: 7

inserisci secondo valore: 3

il numero maggiore e': 7

SOLUZIONE:

```
#include <stdio.h>  
  
int main(){  
    int x;  
    int y;  
  
    printf("inserisci primo valore: ");  
    scanf("%d" , &x);  
  
    printf("inserisci secondo valore: ");  
    scanf("%d" , &y);  
  
    if(x>y){  
        printf("il numero maggiore e': %d", x);  
    }  
    else if(x<y){  
        printf("il numero maggiore e': %d", y);  
    }  
    else{  
        printf("i numeroi sono uguali");  
    }  
    return 0;  
}
```

ESERCIZIO 3:

realizzare un programma che chiede all'utente 1 valore, se il valore è:

$x > 1 \wedge x < 5 \rightarrow$ il voto è estremamente insufficiente

$x \geq 5 \wedge x < 6 \rightarrow$ il voto è insufficiente

$x \geq 6 \wedge x < 7 \rightarrow$ il voto è sufficiente

$x \geq 7 \wedge x < 8 \rightarrow$ il voto è buono

$x \geq 8 \wedge x < 9 \rightarrow$ il voto è ottimo

$x \geq 9 \wedge x \leq 10 \rightarrow$ il voto è eccellente

$x < 0 \vee x > 10 \rightarrow$ il voto non è valido

Esempio Output:

inserisci valore: 7

il voto e' buono

SOLUZIONE:

```
#include <stdio.h>

int main(){

    float val1;

    printf("inserisci valore: ");
    scanf("%f" , &val1);

    if(val1>1 && val1<5){
        printf("il voto e' estremamente insufficiente");
    }
    else if(val1>=5 && val1<6){
        printf("il voto e' insufficiente");
    }
    else if(val1>=6 && val1<7){
        printf("il voto e' sufficiente");
    }
    else if(val1>=7 && val1<8){
        printf("il voto e' buono");
    }
    else if(val1>=8 && val1<9){
        printf("il voto e' ottimo");
    }
    else if(val1>=9 && val1<=10){
        printf("il voto e' eccellente");
    }
    else if(val1<0 || val1>10){
        printf("il voto non e' valido");
    }
}
```

```
    else {  
        printf("errore...");  
    }  
  
    return 0;  
}
```