

Istruzioni di selezione

Come facciamo a scegliere tra più possibilità?

Prendere una decisione significa essere in grado di eseguire un blocco di codice al verificarsi di una determinata condizione

Gli operatori relazionali assumono importanza dal fatto che consentono di prendere decisioni in base al loro risultato

Prendere decisioni

Istruzione if

```
if (condizione)  
{  
    //blocco di codice  
}
```

La condizione è un test in cui vengono messi in relazione due operandi tramite un operatore relazionale o logico

Le istruzioni all'interno del blocco vengono eseguite **solo se** la condizione risulta verificata, ossia vale *true*

Istruzione if

Esempio

```
//creo ed inizializzo due variabili
```

```
numero = 25;
```

```
numero2 = 52;
```

```
//devo prendere una decisione in base ad una co
```

```
if (numero > 50){
```

```
    cout << "Numero maggiore di 50" << endl;
```

```
}
```

```
//devo prendere una decisione in base ad una co
```

```
if (numero2 > 50){
```

```
    cout << "Numero2 maggiore di 50" << endl;
```

```
}
```

Esercizi

01

Data in input l'età, decidere se si è maggiorenni oppure meno, stampando il risultato

02

Dati in input due numeri, decidere quali dei due è maggiore, stampando il risultato

Istruzione If...else

Se la condizione è vera, allora eseguiamo il codice. Ma se è falsa, cosa possiamo fare?

```
If (condizione){  
    //1° codice di blocco  
} else {  
    //2° codice di blocco  
}
```

Istruzione If ...else

01

Usando l'istruzione *else*, riusciamo ad eseguire un pezzo di codice anche se la condizione non è verificata

02

Quindi:

- Se condizione è vera → eseguo il 1° blocco di codice
- Se condizione è falsa → eseguo il 2° pezzo di codice



esempio

```
//creo ed inizializzo due variabili
numero = 25;
numero2 = 52;

//devo prendere una decisione in base ad una condizione
if (numero > 50){
    cout << "Numero maggiore di 50" << endl;
} else {
    cout << "Numero minore o uguale a 50" << endl;
}

//devo prendere una decisione in base ad una condizione
if (numero2 > 50){
    cout << "Numero2 maggiore di 50" << endl;
} else {
    cout << "Numero2 minore o uguale a 50" << endl;
}
```

Esercizi

Aggiungi un'azione da eseguire
(blocco di codice) ai programmi
precedenti nel caso in cui la
condizione sia falsa

Istruzione switch

E se la nostra condizione potesse assumere solo determinati valori, e per ognuno di essi dovessimo scrivere un pezzo di codice?

```
switch (variabile di controllo){  
    case valore1:  
        istruzione1;  
        break;  
  
    ....  
  
    case valoren:  
        istruzionen;  
        break;  
  
    default: istruzione;  
  
}
```

Istruzione switch

01

La variabile di controllo è quella responsabile della decisione, per cui il programma andrà a guardare il valore di tale variabile ed esegue un pezzo di codice in base al suo valore

02

Per ogni *case*, dobbiamo specificare il *valore* e le istruzioni da eseguire

03

Una volta terminato il *case*, usare il *break* per uscire da tale condizione

esempio

```
//creo ed inizializzo le variabili
int numero;

//mi faccio dare il numero che andrà a diventare la mia condizione
cout << "Inserisci un numero intero da 1 a 7" << endl;
cin >> numero;

//tramite lo switch eseguo un pezzo di codice specifico in base al
switch (numero){
    case 1:
        cout << "Oggi e' lunedì" << endl;
        break;
    case 2:
        cout << "Oggi e' martedì" << endl;
        break;
    case 3:
        cout << "Oggi e' mercoledì" << endl;
        break;
    case 4:
        cout << "Oggi e' giovedì" << endl;
        break;
    case 5:
        cout << "Oggi e' venerdì" << endl;
        break;
    case 6:
        cout << "Oggi e' sabato" << endl;
        break;
    case 7:
        cout << "Oggi e' domenica" << endl;
        break;
    default:
        cout << "Oggi e' il giorno giusto per dormire" << endl;
        break;
}
```

Esercizio

Usare il costrutto
switch per definire
in quale mese
siamo

Tempo di pratica

