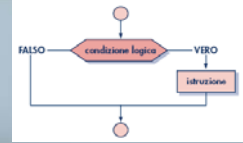


Linguaggio C++

Selezione

Struttura di controllo decisionale

- Ogni linguaggio di programmazione presenta almeno una **struttura di controllo decisionale**.
- La sintassi del linguaggio C è la seguente.
`if (<espressione>)
<istruzione>;`
- Si tratta di una **selezione**, all'interno della quale se e solo se **espressione** è vera viene eseguita l'istruzione.

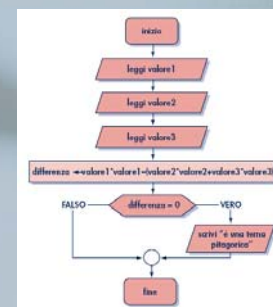


Programmazione: IF
`if (<espressione>)
<istruzione>;`

Problema

- **Problema:** *verificare se i tre valori passati in ingresso sono una terna pitagorica.*
- **Nota:** il primo valore immesso deve essere il maggiore dei tre.
- **Input:** tre valori numerici interi, il primo deve essere il maggiore dei tre.
- **Output:** in caso di verifica positiva, viene segnalato che si tratta di una terna pitagorica.

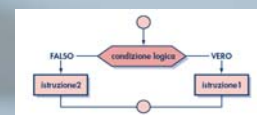
Algoritmo



Programma in C++

- Scrivere il programma equivalente in linguaggio C++
- ricordando che:
 - il confronto fra due valori avviene mediante gli operatori logici:
 - <, <=, ==, >, >=, !=
 - l'istruzione condizionata va indentata facendola rientrare di due spazi rispetto al resto del programma
- es:
`a=3;
b=2;
if (a==b)
 cout << "le variabili hanno lo stesso valore";
cout << "fine programma";`

if ... else



Programmazione: If ... Else in C
`if (<espressione>)
<istruzione1>;
else
<istruzione2>;`

un esempio

```
...
if (differenza==0)
    cout << "e' una terna pitagorica";
else
    cout << "non e' una terna pitagorica";
```

Problema

Problema: verificare se un numero intero è pari o dispari.
Input: un numero intero.
Output: scrive se il numero ricevuto in ingresso è pari o dispari.

Programma risolutivo

```
...
main()
{
    int numero;
    cout << "Verifica se un numero e' pari o dispari";
    cout << endl;
    cout << "Immetti un numero intero: ";
    cin >> numero;
    if((numero % 2) == 0)
        cout << "e' pari";
    else
        cout << "e' dispari";
}
```

Operatori aritmetici

Operatore	Descrizione	Esempio	Descrizione esempio
-	Negazione unaria	-x	Il valore di x cambiato di segno
*	Prodotto	x*y	x moltiplicato y
/	Divisione	x/y	x diviso y
%	Resto della divisione intera	x%y	Resto della divisione intera di x per y
+	Addizione	x+y	x sommato a y
-	Sottrazione	x-y	x meno y

Operatori di confronto

Operatori di confronto	Descrizione	Esempio	Descrizione esempio
>	maggiore	x>y	x maggiore di y
<	minore	x<y	x minore di y
>=	maggiore uguale	x>=y	x maggiore uguale di y
<=	minore uguale	x<=y	x minore uguale di y
==	uguale	x==y	x uguale a y
!=	diverso	x!=y	x diverso da y

Operatori logici

Operatori logici	Descrizione	Esempio	Descrizione esempio
!	NOT logico	!x	NOT x
&&	AND logico	x&& y	x AND y
	OR logico	x y	x OR y

Blocco di istruzioni

■ un **bloccoIstruzioni** è costituito da un insieme d'istruzioni qualsiasi raccolte tra parentesi graffe:

```
if(<espressione>)
{
    <istruzione1>;
    <istruzione2>;
    ...
    <istruzioneN>;
}
```

Programmazione: If ... [Else]

```
if(<espressione>)
    <bloccoIstruzioni>
[else <bloccoIstruzioni>]
```

Problema

- Scrivere un programma che riceve in input un valore intero compreso fra 0 e 100.000 e fornisce come risultato il numero di banconote da 500€ da 200€ 100€ da 50€ da 10€ e di monete da 2€ e da 1€ che servono per raggiungere l'importo.
- Suggerimento: utilizzare l'operatore %.

Problema

- Scrivere un programma che riceve 4 valori e controlla se possono essere i lati di un rettangolo, in caso affermativo visualizza l'area.