

## Condizioni annidate e operatori logici

Tradurre la seguente frase (condizione), in particolare le **parole evidenziate**, in modo da spiegare cosa si deve operativamente verificare

se **almeno uno** dei **voti è insufficiente**  
**allora** visualizzare un messaggio in output

va spiegato che

voto insufficiente significa:  $\text{voto} < 6$

almeno uno dei voti significa. un voto oppure un altro voto

se  $\text{voto informatica} < 6$  **oppure**  $\text{voto economia} < 6$  **oppure**  $\text{voto italiano} < 6$   
**allora** visualizzare un messaggio di output

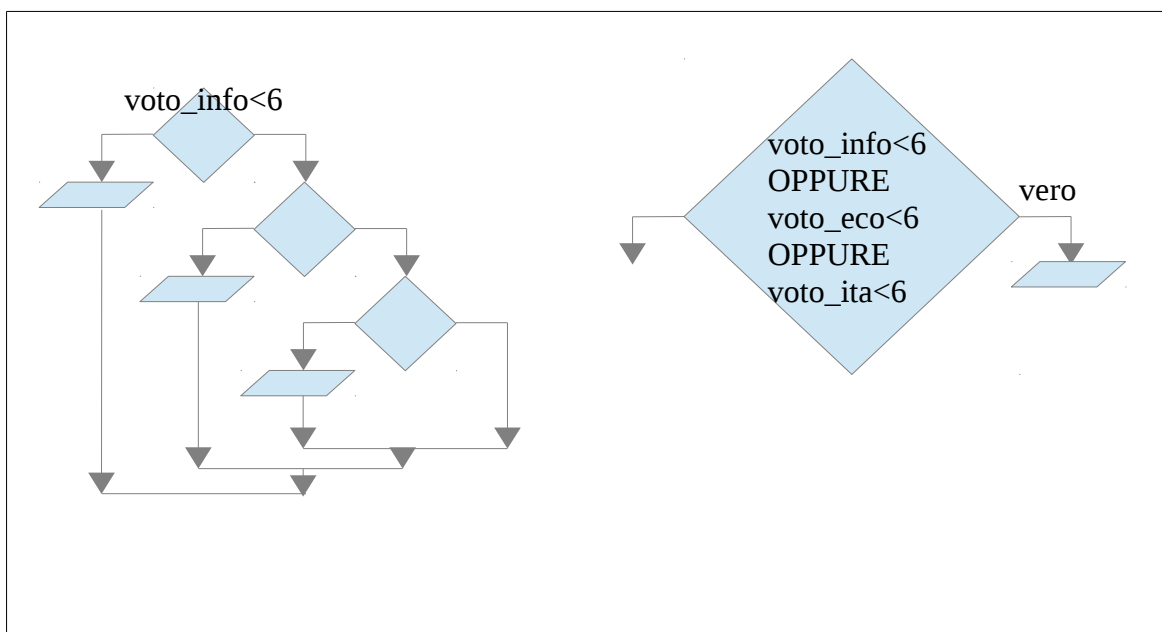
tabella (tabella di verità)

| voto info | voto eco | voto ita | output |
|-----------|----------|----------|--------|
| 6         | 6        | 6        | no     |
| 5         | 6        | 6        | sì     |
| 5         | 5        | 6        | sì     |
| 5         | 5        | 5        | sì     |
| ...       | ...      | ...      | ...    |

OSSERVAZIONE. Usando la congiunzione **oppure** la condizione è verificata sempre, eccetto il caso in cui tutti i voti siano sufficienti.

Quindi, il precedente problema potrebbe essere risolto in due modi:

usando tre condizioni annidate, oppure usando una sola condizione composta con OPPURE



In italiano le diverse condizioni si potevano unire usando anche la congiunzione E, ma il significato sarebbe stato completamente diverso. Possiamo confrontare le differenze usando due tabelle di verità (output O, output E):

**se** voto informatica < 6 **e** voto economia < 6 **e** voto italiano < 6  
**allora** visualizzare un messaggio di output

tabella di verità

| voto info | voto eco | voto ita | output O | output E |
|-----------|----------|----------|----------|----------|
| 6         | 6        | 6        | no       | no       |
| 5         | 6        | 6        | sì       | no       |
| 5         | 5        | 6        | sì       | no       |
| 5         | 5        | 5        | sì       | sì       |
| ...       | ...      | ...      | ...      | ...      |

OSSERVAZIONE. Usando la congiunzione **e** la condizione non è mai verificata , eccetto il caso in cui tutti i voti siano insufficienti

Quelle che in italiano si chiamano congiunzioni, in informatica vengono detti OPERATORI LOGICI e possono essere tradotti in un linguaggio di programmazione.

## Operatori di confronto

Quando si vuole verificare una condizione è necessario eseguire un'operazione di confronto, come quelle nei seguenti:

| esempio | descrizione                |
|---------|----------------------------|
| voto<6  | voto minore di sei         |
| voto>=6 | voto maggiore o pari a sei |
| voto==6 | voto pari a sei            |
| voto!=6 | voto diverso da sei        |

## Algebra di Boole

L'algebra fu sviluppata nel 1800 da George Boole (detta algebra booleana) per definire la logica delle *proposizioni*, dove si studia lo stato (il valore) di verità delle proposizioni. Una proposizione può essere *vera* oppure *falsa*.

ATTENZIONE: non confondere in italiano il termine *proposizione* con *preposizione*, che ha un altro significato...

Esempio di proposizione (con valore tra parentesi):

- P = “il quadrato ha 4 lati” (vero)
- Q = “il tuo cane miagola” (falso)

Nell'algebra di Bool esistono diversi operatori (detti operatori booleani oppure connettivi logici) che possono essere applicati ad una o più proposizioni.

- AND (detto congiunzione)
- OR (detto disgiunzione)
- NOT (detto negazione)
- altri meno famosi come: NAND, NOR, XOR,...

### Esempio di NOT

Applicando l'operatore NOT alle due precedenti proposizioni il loro valore di verità diventa l'opposto:

- NOT P = “il quadrato NON ha 4 lati” (falso)
- NOT Q = “il tuo cane NON miagola” (vero)

### Esempio di AND

Applicando l'operatore AND, si ottiene una nuova proposizione il cui valore di verità dipende dalla verità di entrambe le proposizioni di partenza (è vera solo se sono entrambe vere)

- P AND Q = “il quadrato ha 4 lati” E “il tuo cane miagola” (falso)

### Esempio di OR

Applicando l'operatore OR, si ottiene una nuova proposizione il cui valore di verità dipende dalla verità di entrambe le proposizioni di partenza (è vera se almeno una delle due è vera):

- P OR Q = “il quadrato ha 4 lati” OPPURE “il tuo cane miagola” (vero)

### In Javascript (e altri linguaggi simili)

Ecco come si traducono nel linguaggio di programmazione

| operatore | Javascript | ESEMPIO    |
|-----------|------------|------------|
| AND       | &&         | x<9 && y<0 |
| OR        |            | x<9    y<0 |
| NOT       | !          | !(x<9)     |

## La tabella di verità

Per ogni operatore booleano (AND, OR, NOT, ecc.) si può costruire una tabella che aiuta a descriverlo. Nella seguente tabella sono messe a confronto gli operatori AND e OR. Le prime due colonne sono due proposizioni su cui applicare tali operatori.

| P | Q | P and Q | P or Q |
|---|---|---------|--------|
| V | V | V       | V      |
| V | F | F       | V      |
| F | V | F       | V      |
| F | F | F       | F      |

## Il calcolo automatico

Dopo circa un secolo dalla sua ideazione, l'algebra di George Boole ha trovato applicazione anche nell'elaboratore. Poiché l'elaboratore utilizza il sistema di numerazione in base 2, al valore "vero" (true) viene fatto corrispondere il valore 1 mentre al "falso" (false) lo 0.

Più in generale, si può dire che qualsiasi valore diverso da 0 è considerato vero, infatti, in quasi tutti i linguaggi di programmazione il numero intero ZERO viene interpretato come "falso".

Il programmatore può scrivere espressioni che hanno un risultato numerico, come  $x=33*4$ ; o espressioni che hanno un risultato "vero" o "falso", come  $x>0$ .

Alle espressioni di confronto, come  $a<b$ , possono essere applicati i precedenti operatori logici. Il calcolo automatico di queste espressioni (come  $a<b$ ) permette al programma di eseguire un'istruzione oppure un'altra a seconda se l'espressione venga valutata "vera" oppure "falsa".

Nelle istruzioni di scelta condizionata sarebbe opportuno usare solo il secondo tipo, ma questa non è una regola fissa.

Quando vengono usate le espressioni che producono un risultato numerico, il numero viene automaticamente convertito in "falso" o "vero". Se vale zero è "falso", se è diverso da zero è "vero".

## Attenzione

In generale una condizione si può descrivere sempre in due modi, entrambi corretti, come nei seguenti esempi. Per esercizio, trovare quali sono le differenze.

