



**POLITECNICO**  
MILANO 1863

*Esercitazione 1:*

*esercizi con Algebra di Boole, IF e  
WHILE*

*Ing. Andrea Mazzoleni*

# Richiamo, Tabelle di Verità

| a | b | a && b | a    b | ~a | xor(a,b) |
|---|---|--------|--------|----|----------|
| 0 | 0 | 0      | 0      | 1  | 0        |
| 0 | 1 | 0      | 1      | 1  | 1        |
| 1 | 0 | 0      | 1      | 0  | 1        |
| 1 | 1 | 1      | 1      | 0  | 0        |

Or esclusivo: vero quando è vera solo uno delle due espressioni coinvolte  $Xor(a,b) == a || b \&\& (\sim(a \text{ AND } b))$

# Operatori Logici: Forma Generale

Operatori binari: **AND** (&&, oppure &, oppure and), **OR** (||, oppure |, oppure or), **XOR** (xor):

a OP1 b      per la notazione simbolica  
OP(a,b)      per la notazione testuale

Operatori unari: NOT (~):

OP2 a

a,b possono essere variabili, costanti, espressioni da valutare, scalari o vettori (dimensioni compatibili)

Valori numerici di a, b vengono interpretati come logici:

- 0 come falso
- tutti i numeri diversi da 0 come vero

Ordine Operatori Logici in assenza di parentesi (elementi a priorità maggiore in alto):

1. negazione (NOT)  $\sim$
2. operatori di relazione  $<$ ,  $>$ ,  $<=$ ,  $>=$
3. uguaglianza  $==$ , disuguaglianza  $\neq$ ,
4. congiunzione (AND)  $\&\&$
5. disgiunzione (OR)  $||$

*Esempio*

–  $x > 0 || y == 3 \&\& \sim(z > 2)$

–  $(x > 0) || ((y == 3) \&\& \sim(z > 2))$

# Aritmetica degli Operatori Logici

Gli operatori `&&` e `||` sono commutativi

$$- (a \ \&\& \ b) \ == \ (b \ \&\& \ a)$$

$$- (a \ || \ b) \ == \ (b \ || \ a)$$

Le doppie negazioni si elidono:  $\sim\sim a \ == \ a$

Rappresenta tutti i possibili modi di valutare un' espressione booleana composta

Una riga per ogni possibile assegnamento di valori logici alle variabili:

- $n$  variabili logiche (espressioni booleane)  $\rightarrow 2^n$  possibili assegnamenti, quindi  $2^n$  righe.

Una colonna per ogni espressione che compone l'espressione data (inclusa la formula stessa)

# Esempio Tabella di Verità

Compilare la tabella di verità della seguente espressione:

**A && ~B || C**

# Esempio Tabella di Verità

**A && ~B || C**

| A | B | C |  |  |  |
|---|---|---|--|--|--|
|   |   |   |  |  |  |
|   |   |   |  |  |  |
|   |   |   |  |  |  |
|   |   |   |  |  |  |
|   |   |   |  |  |  |
|   |   |   |  |  |  |
|   |   |   |  |  |  |
|   |   |   |  |  |  |
|   |   |   |  |  |  |



# Altro Esempio di Tabella di Verità

**A && (~B || C)**

# Altro Esempio di Tabella di Verità

$A \ \&\& \ (\sim B \ || \ C)$

| A | B | C |  |  |  |
|---|---|---|--|--|--|
|   |   |   |  |  |  |
|   |   |   |  |  |  |
|   |   |   |  |  |  |
|   |   |   |  |  |  |
|   |   |   |  |  |  |
|   |   |   |  |  |  |
|   |   |   |  |  |  |
|   |   |   |  |  |  |

# Operatori Logici: Leggi di de Morgan

Leggi di De Morgan: illustrano come distribuire la negazione rispetto a `||` e `&&`

$$1. \sim(a \ \&\& \ b) == \sim a \ || \ \sim b$$

$$2. \sim(a \ || \ b) == \sim a \ \&\& \ \sim b$$

Es:  $\sim((a \geq 5) \ \&\& \ (a \leq 10)) \rightarrow$  [De Morgan]

$\sim(a \geq 5) \ || \ \sim(a \leq 10) \rightarrow$  [proprietà  $\geq$  e  $\leq$ ]

$\sim\sim(a < 5) \ || \ \sim\sim(a > 10) \rightarrow$  [doppia negazione]

$((a < 5) \ || \ (a > 10))$

Dimostrare che le seguenti espressioni sono equivalenti

$$- A \quad || \quad C \quad \&\& \quad \sim B$$

$$- \sim ( (B \quad || \quad \sim C) \quad \&\& \quad \sim A )$$

Due possibili soluzioni:

- Applicando le leggi di De Morgan cerco di passare da una all'altra
- Calcolo entrambe le tabella di verità e mostro che coincidono

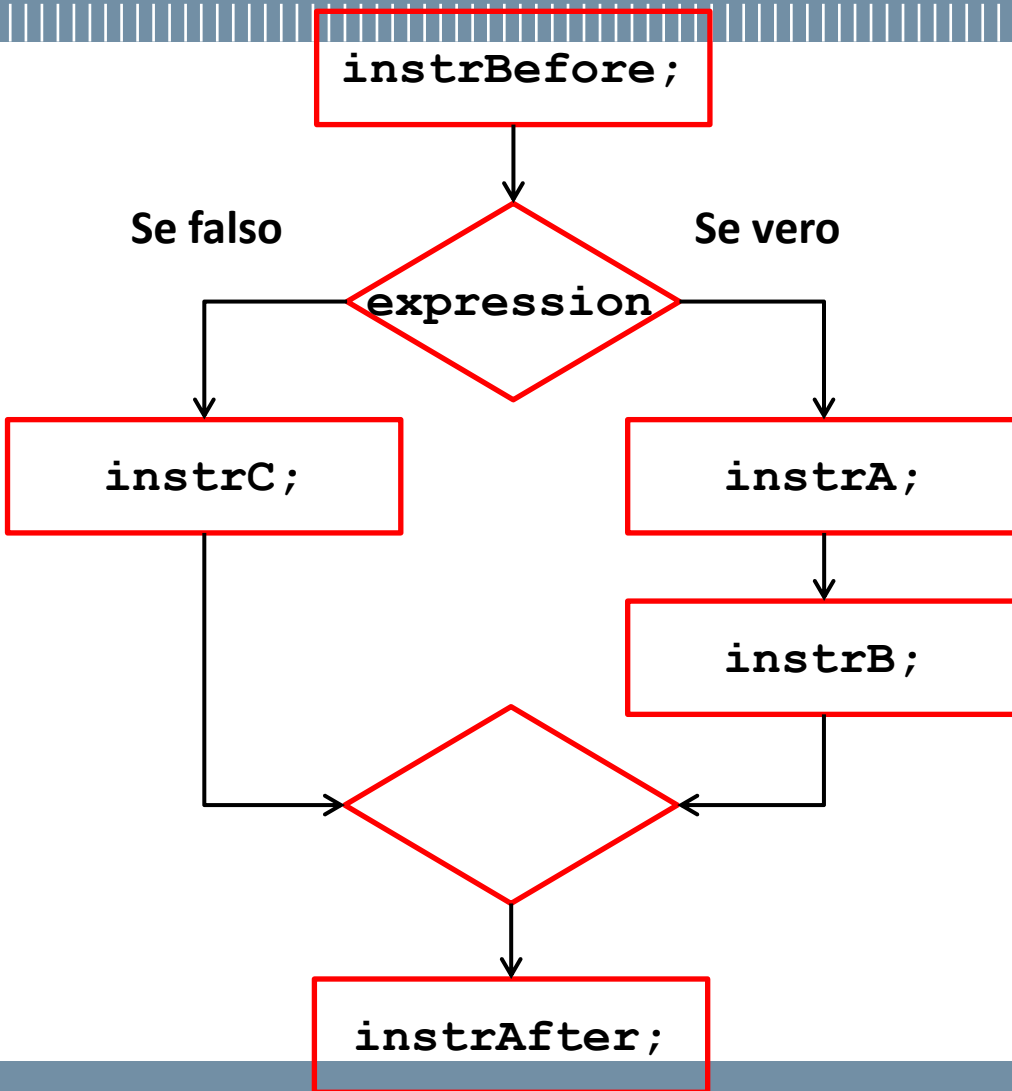
Dimostrare che le seguenti espressioni sono equivalenti

$$- A \ || \ C \ \&\& \ \sim B$$

$$- \sim ( (B \ || \ \sim C) \ \&\& \ \sim A)$$

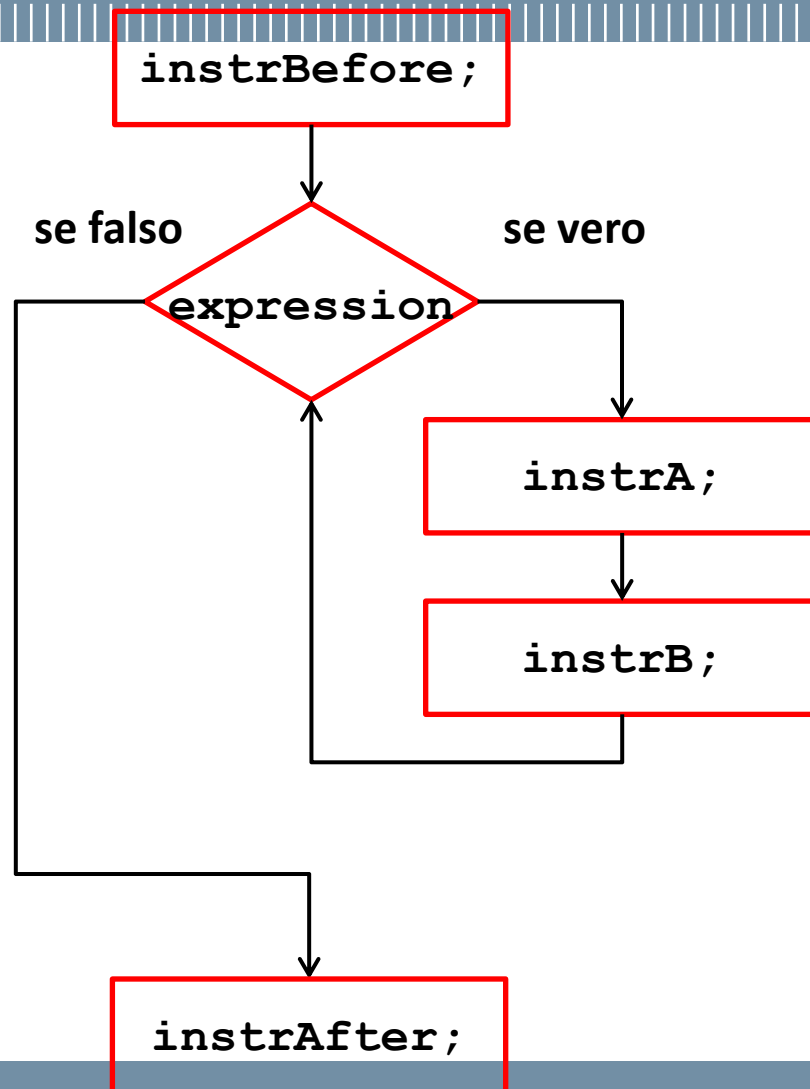
- $\sim ( (B \ || \ \sim C) \ \&\& \ \sim A)$

# Ripasso costruito IF



```
instrBefore;  
if expression  
    instrA;  
    instrB;  
else  
    instrC;  
end  
instrAfter;
```

# Ripasso ciclo While



```
instrBefore;  
while expression  
    instrA;  
    instrB;  
end  
instrAfter;
```

# Programma 1: Anno bisestile (Boole)

- Scrivere uno script MATLAB che permette all'utente di inserire un anno.
- Il programma verifica se tale anno è BISESTILE o meno e stampa un opportuno messaggio.

Si ricorda che un anno è bisestile se è multiplo di 4 ma non di 100 oppure è multiplo di 400.

Hint: usare condizione booleana



## Programma 2: Successione di Fibonacci

Scrivere un programma che stampa a video la successione di Fibonacci.

Hint:

La successione di Fibonacci è una sequenza di numeri dove, a partire dal terzo, ogni numero è la somma dei due precedenti.

Hint: Usare un ciclo WHILE

Es: 0 1 1 2 3 5 8 13 ...

## Programma 3: Minimo comune multiplo

Scrivere un programma per calcolare il minimo comune multiplo di due numeri.

Hint:

Usare ciclo WHILE

Usare funzione 'mod'

# Programma 4: Output grafici

- Triangolo basso

```
+
+ +
+ + +
+ + + +
+ + + + +
+ + + + + +
+ + + + + + +
+ + + + + + + +
+ + + + + + + + +
```

- Triangolo alto

```
+ + + + + + + + + +
+ + + + + + + + +
+ + + + + + + +
+ + + + + + +
+ + + + + +
+ + + + +
+ + + +
+ + +
+ + +
+ +
+
```

- Scacchiera

```
£ € £ € £ € £ € £
€ £ € £ € £ € £ €
£ € £ € £ € £ € £
€ £ € £ € £ € £ €
£ € £ € £ € £ € £
€ £ € £ € £ € £ €
£ € £ € £ € £ € £
€ £ € £ € £ € £ €
£ € £ € £ € £ € £
€ £ € £ € £ € £ €
```