

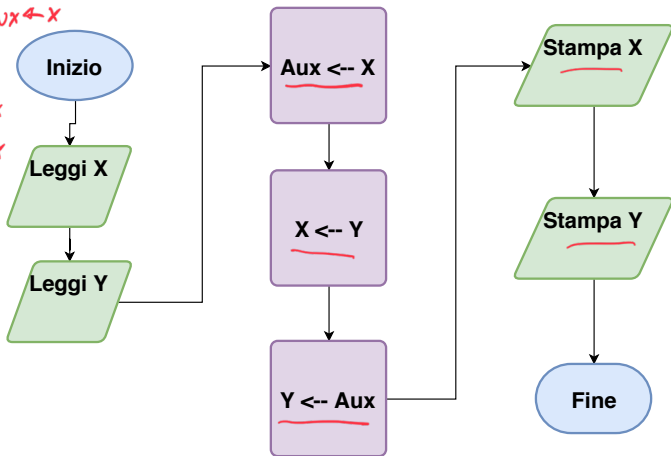
Diagrammi di flusso

Esempio: Scambio di due numeri



Handwritten notes on the left side:

- Leggi X
- Leggi Y
- $\rightarrow X \leftarrow Y$
- $\rightarrow Y \leftarrow \text{AUX}$
- X: 2 (in a red box)
- Y: 3 (in a red box)
- AUX: (empty red box)
- 2 (in a red box)
- 3 (in a red box)
- 2 (in a red box)
- 3 (in a red box)
- 2 (in a red box)



Istruzioni condizionali

Ci sono dei momenti in cui il **flusso** di esecuzione può scegliere tra **diverse direzioni**.

In genere, questi **salti** sono subordinati al verificarsi di una **condizione** (che può risultare vera o falsa);

⇒ Istruzione condizionale

Proposizione

Costrutto linguistico del quale si può affermare la **veridicità**.

ES: Il numero 2 é pari \implies VERO

Il valore di verità di una proposizione è il suo essere vera o falsa.

Predicato

Una proposizione il cui valore di verità dipende dall'istanza-
zione di alcune variabili

ES: La variabile X è minore di 30. / La variabile Y è maggiore
della variabile Y

Valutazione di un predicato

Operazione che permette di **determinare** se il predicato é vero o falso, sostituendo alle variabili i loro valori attuali.
I valori VERO e FALSO sono **valori logici o booleani**

Operatori relazionali

Esprimere in modo conciso i predicati mediante variabili e operatori relazionali:

= (uguale)

\neq (diverso)

\leq (minore o uguale)

\geq (maggiore o uguale)

$<$ (minore)

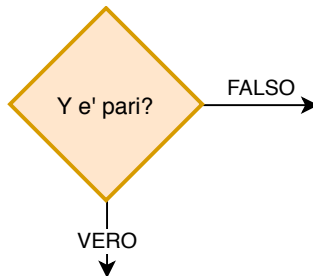
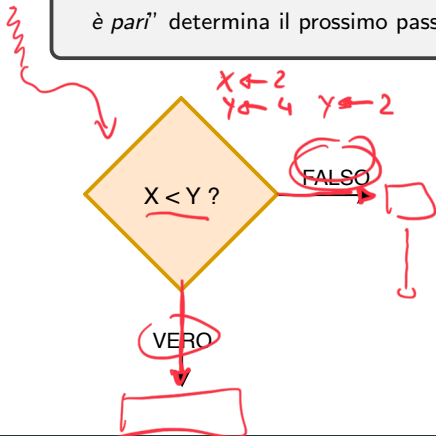
$>$ (maggiore)

Diagrammi di flusso

Blocco di istruzione condizionale



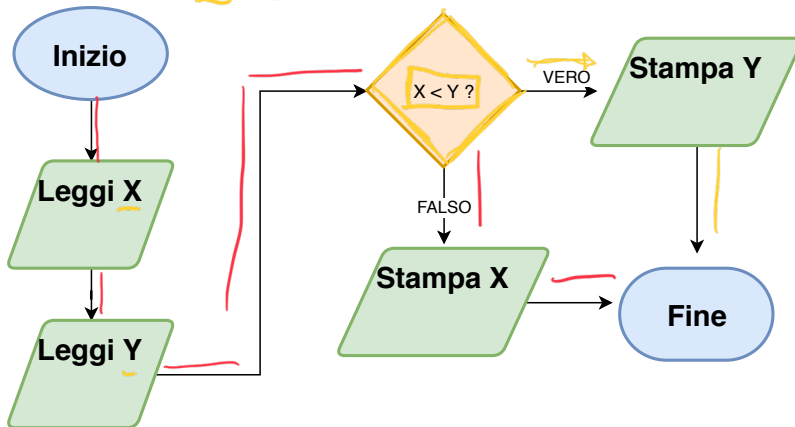
Il valore di verità del predicato " $X < Y$ " (a sinistra) o del predicato " Y è pari" determina il prossimo passo nel flusso di esecuzione



Diagrammi di flusso

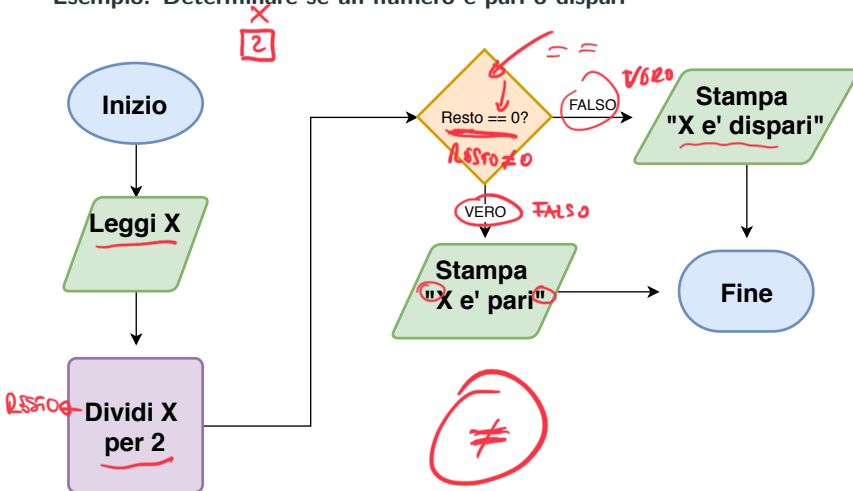
Esempio: stampare il massimo tra due numeri

X
4 Y
6



Diagrammi di flusso

Esempio: Determinare se un numero è pari o dispari

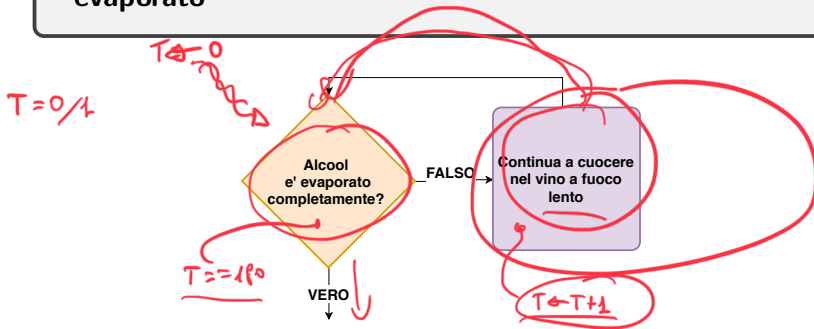


Diagrammi di flusso

Cicli (loop) o ripetizioni

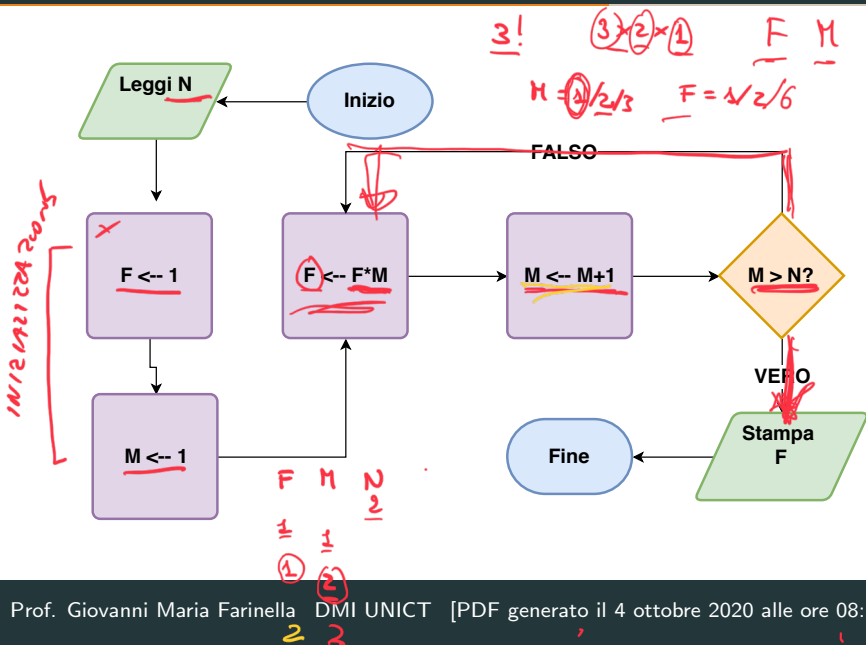
Loop o ciclo: sfumare con vino bianco

(dopo aver versato mezzo bicchiere di vino bianco ...) far sobbollire a fuoco lento **fino a quando tutto l'alcool sarà evaporato**



Notazione Lineare Strutturata (NLS)

Esempio: calcolo del fattoriale di un numero





Se gli algoritmi da rappresentare sono **articolati e complessi**, i diagrammi di flusso a blocchi possono riverlarsi:

- poco pratici → soggetti ad errori
- poco leggibili

Alternativa: NLS (Notazione Lineare Strutturata)

Notazione lineare strutturata

Costrutti

Sequenza

Equivalente ad uno o più blocchi di operazioni che si susseguono.

Selezione

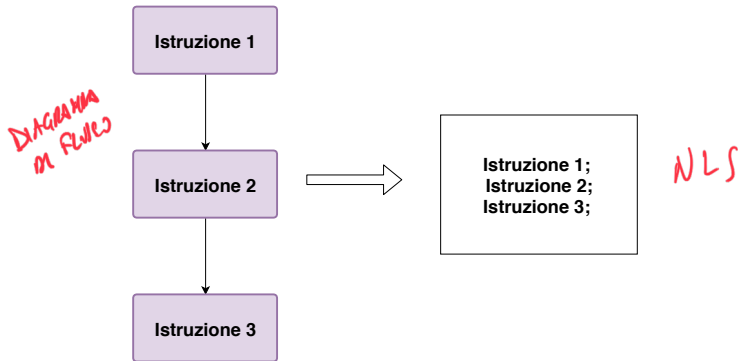
Equivalente al blocco condizionale

Iterazione

Equivalente al blocco condizionale più uno o più blocchi di operazioni disposti in modo da formare un ciclo

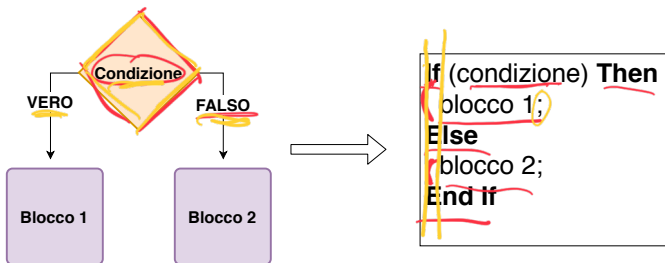
Notazione lineare strutturata

Sequenza



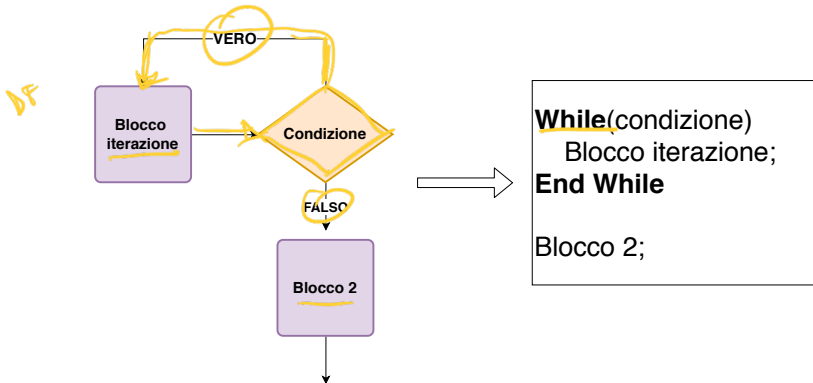
Notazione lineare strutturata

Selezione



Notazione lineare strutturata

Iterazione



Teorema di Böhm-Jacopini (1966)

Ogni algoritmo può essere costruito utilizzando unicamente tre strutture (o schemi di controllo):

- la sequenza
- la selezione
- il ciclo o iterazione



⇒ Ogni altro tipo di istruzione può essere **sostituito da una combinazione dei tre schemi precedenti**



In un qualsiasi linguaggio di programmazione sono **sufficienti espressioni che rappresentino le tre strutture NLS** per scrivere e implementare qualsiasi programma.



Ovviamente i **linguaggi di programmazione** mettono a disposizione anche **altri costrutti** (ad esempio il costrutto `for` del C/C++).

NLS: esempi

```
1 Inizio  
2   Leggi X;  
3   Leggi Y;  
4    $Z \leftarrow X + Y$ ;  
5   Stampa Z;  
6 Fine
```

Massimo tra due numeri

1 **Inizio**

2 Leggi X ;

3 Leggi Y ;

4 **If** (X > Y) then

5 Stampa X ;

6 **Else**

7 Stampa Y ;

8 **End If**

9 **Fine**

IF (X < Y) THEN
 Stampa Y ;
ELSE
 Stampa X ;
ENDIF

Stampa i numeri da 1 a N

1 **Inizio**

N

M

Stampa

2 Leggi N

2

0/1/2

1/2

3 $M \leftarrow 0$ ←

4 **While**($M \leq N$) **Do**

5 $M \leftarrow M + 1$ →

6 Stampa M

7 **End While**

8 **Fine**

Somma dei primi N numeri

```
1  Inizio
2  Leggi N;
3   $i \leftarrow 0;$ 
4   $S \leftarrow 0;$ 
5  While ( $i \leq N$ ) Do
6       $i \leftarrow i + 1;$ 
7       $S \leftarrow S + i;$ 
8  End While
9  Stampa S
10 Fine
```

Handwritten notes:

1 2 3 4
1 0 1 2 3 $i \leftarrow 1;$

Ja Sai
 $i \leftarrow i + 1$

Stampa le prime N+1 potenze del numero 2

P
D

```
1  Inizio
2  Leggi N
3  M ← 0
4  P ← 1
5  While (M ≤ N) do
6      Stampa P
7      P ← P · 2
8      M ← M + 1
9  End While
10 Fine
```



```

1  Inizio
2  Leggi A, B
3  MA ← A
4  MB ← B
5  While( MA < > MB ) do
6      If( MA > MB ) Then
7          MB ← MB + B
8      Else
9          MA ← MA + A
10     End If
11 End While
12 → Stampa "mcm=" MA
13 Fine
    
```

Es: $\text{mcm}(3,7) = 21$

<u>MA</u>	<u>MB</u>
<u>3</u>	<u>7</u>
<u>6</u>	
<u>9</u>	
	<u>14</u>
12	
15	
	<u>21</u>
18	
<u>21</u>	

Algoritmo di Euclide per il M.C.D.



```
1  Inizio
2  Leggi A, B
3  If (A < B) Then
4  |  MB ← A
5  |  MA ← B
6  Else
7  |  MB ← B
8  |  MA ← A
9  End If
10 While (MB <> 0) do
11   r ← MA % MB
12   MA ← MB
13   MB ← r
14 End While
15 Stampa "MCD=" MA
16 Fine
```

Es: $\text{MCD}(21, 14) = 7$

MA	MB	MA % MB
----	----	---------

<u>21</u>	<u>14</u>	<u>7</u>
8	<u>7</u>	<u>0</u>
<u>7</u>	<u>0</u>	—