



IIS2 INFORMATICA

# Ingegnerizzazione di un Algoritmo col metodo DAB

**Draft version**

rev.0.91 - 21 ott 2023

Ing. Nino Bucca

[WWW.ALGORITMOSTEM.IT](http://WWW.ALGORITMOSTEM.IT)

Appunti in formato bozza, intesi esclusivamente di ausilio alle lezioni, che le integrano nelle descrizioni e nei ragionamenti su quanto viene riportato in queste pagine.

Potete segnalare eventuali errori di trascrizione all'indirizzo email: [algostem@icloud.com](mailto:algostem@icloud.com)

Licenza Creative Commons  
CCBYNCND.

È consentita la condivisione del documento originale a condizione che non venga modificato né utilizzato a scopi commerciali, sempre attribuendo la paternità dell'opera all'autore

# Ingegnierizzazione di un Algoritmo col metodo DAB (diagramma a blocchi)

## Scopo:

- 1) Specifiche sulle prestazioni -  
ottenere un Algoritmo efficace,  
ovvero raggiungere lo scopo prefissato  
(l'algoritmo funzione)  
ed efficiente, ovvero utilizzare il minor  
numero di risorse (l'algoritmo funzione  
con la minore complessità computazionale)
- 2) Specifiche sulle Topologie -  
ottenere un Diagramma a Blocchi  
chiaro e facilmente comprensibile

## ESEMPIO

Sviluppare un algoritmo che  
determini l'appartenenza  
alla stagione estiva, di una  
data correttamente inserita  
in formato numerico gg/mm



L'algoritmo rappresentato nel diagramma a Blocchi è efficace, il risultato dell'elaborazione dei dati è corretto, ma possiamo osservare che:

1) Il controllo  $M > 9$  è ridondante, poiché è dei controlli precedenti, per esclusione, il risultato è sempre vero (reminder: i dati inseriti sono corretti)  
Escludendo tale controllo miglioriamo in efficienza -

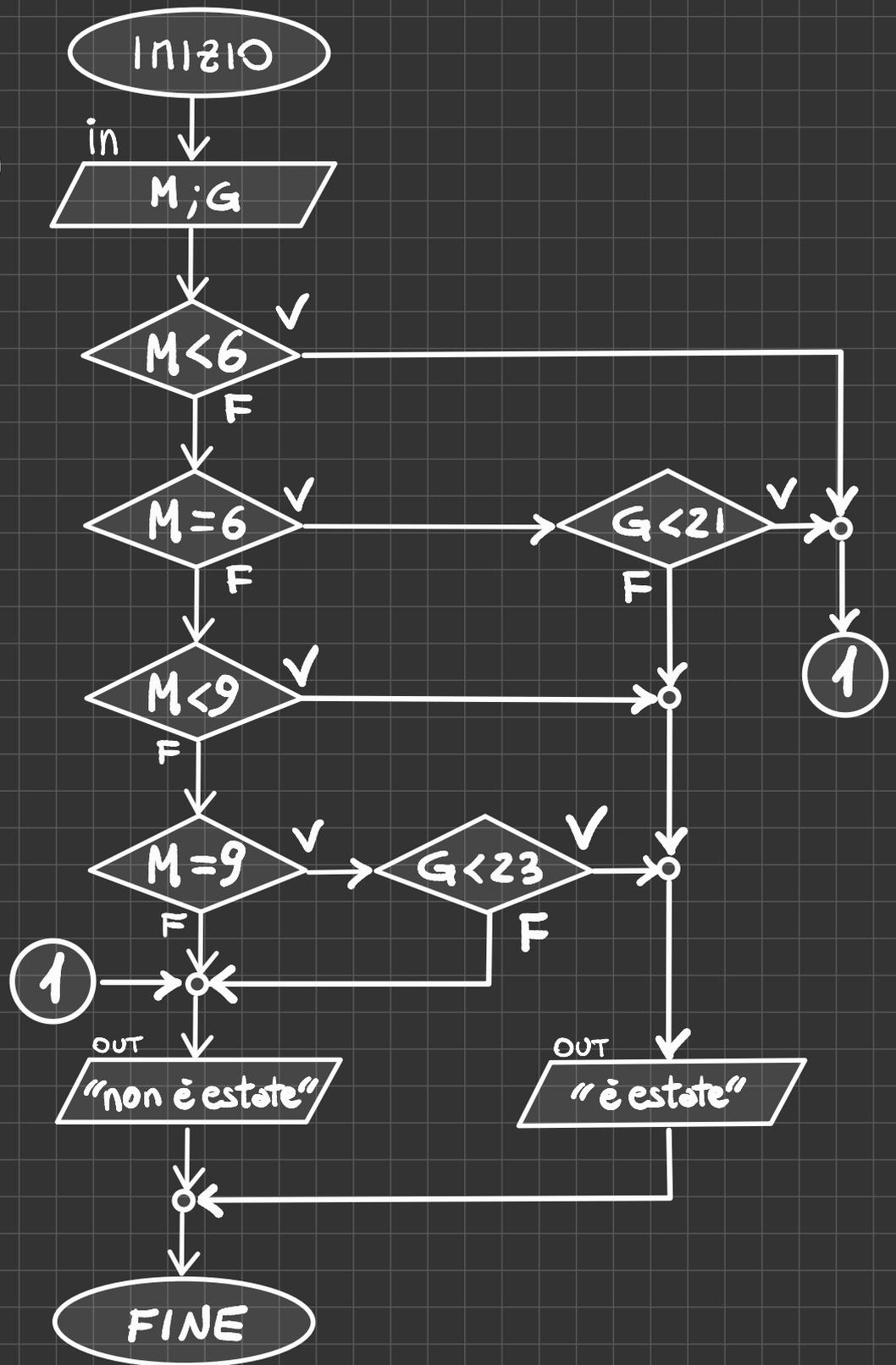
2) Il diagramma a blocchi presenta una ripetizione di blocchi di output equivalenti che possono essere raggruppati.  
Inoltre, bisogna evitare, se possibile, che le linee di flusso si sovrappongano -

Effettuando queste modifiche miglioriamo in chiarezza e leggibilità -

legende

M mese (numero)

G giorno (numero)



Adesso l'algoritmo è più efficiente -  
Possiamo renderlo ancora più efficiente?

Bisogna trovare una differente strategia per ottenere lo stesso risultato

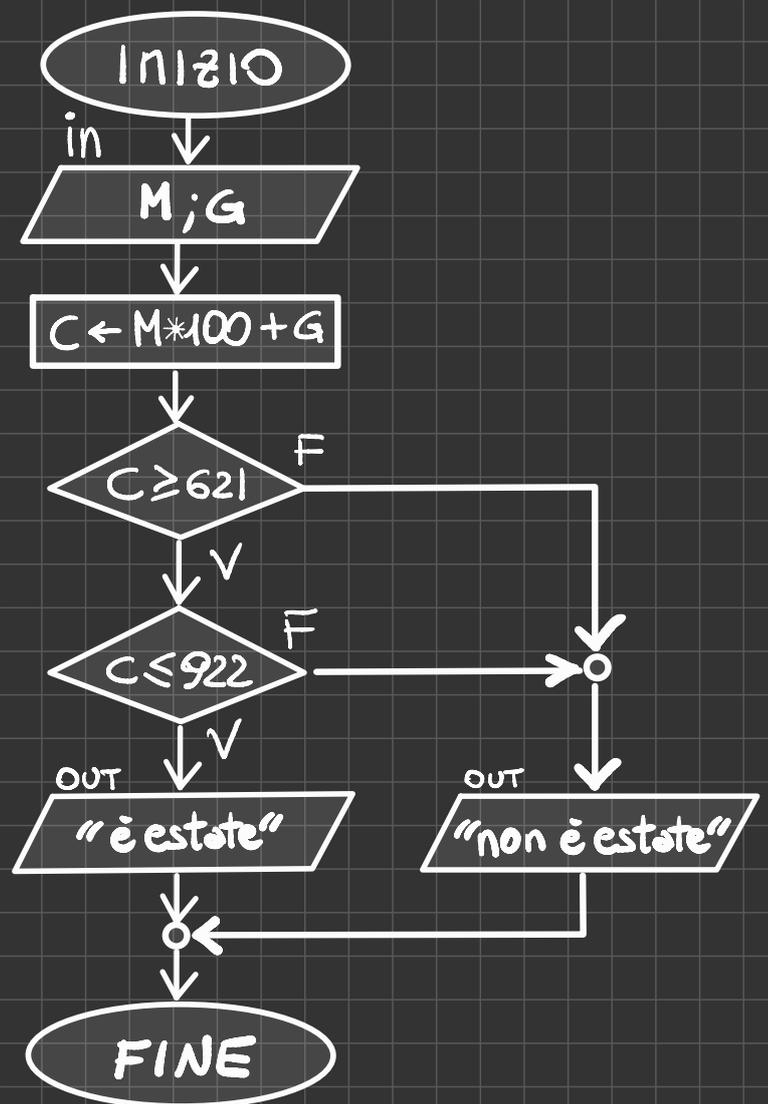
Ad esempio, associando alle date un identificatore che assuma il valore:  $c = \text{mese giorno}$ ,  
(ad esempio 12/05  $\rightarrow c = 0512$ )

i giorni estivi saranno quelli compresi tra  $c = 0621$  (21 giugno) e  $c = 0922$  (22 settembre)

A questo punto è sufficiente controllare che la variabile  $c$  sia compresa tra questi valori ed "il gioco è fatto!"

### legenda

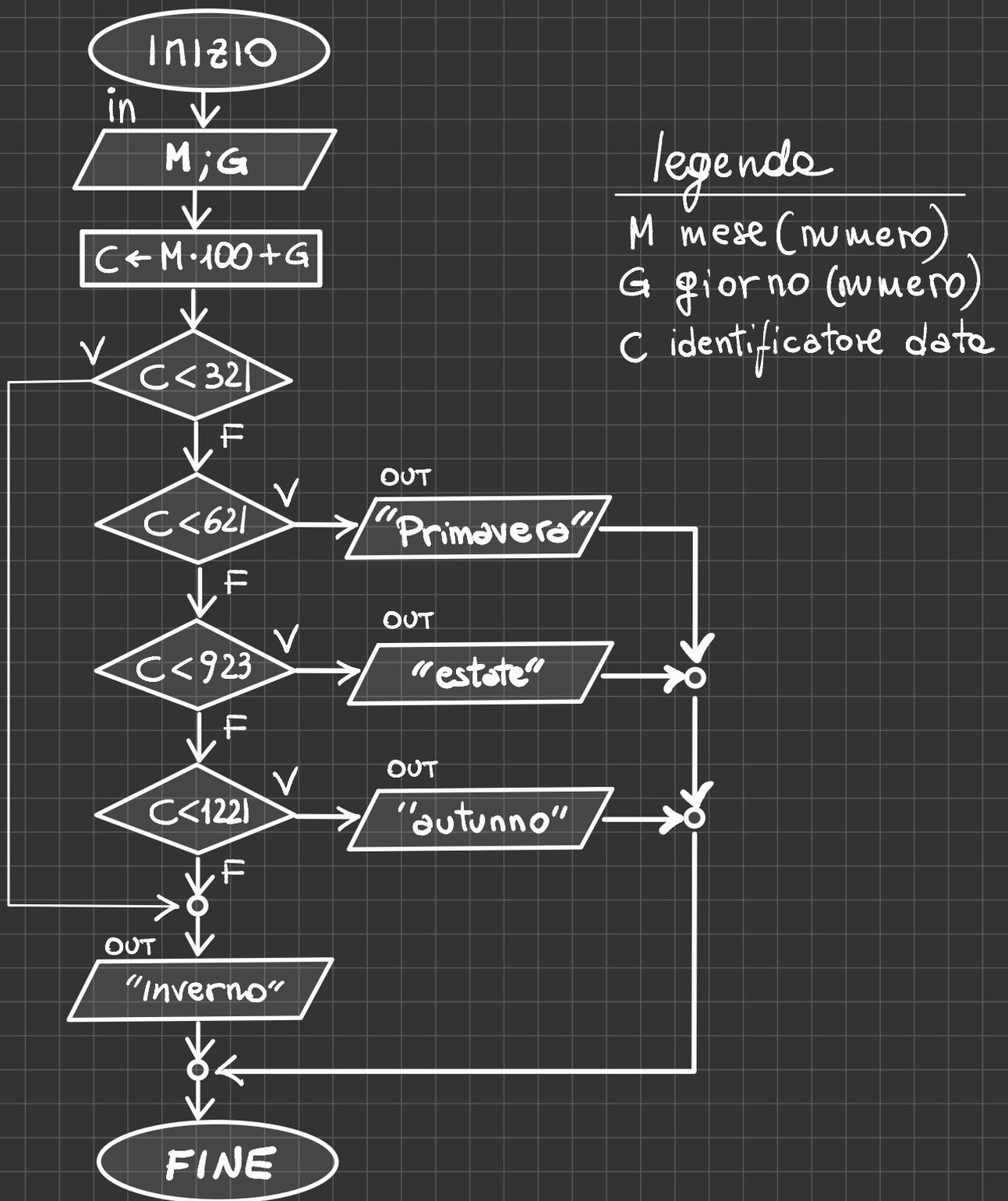
M mese (numero)  
G giorno (numero)  
C identificatore data



Utilizzando la stessa tecnica, è possibile estendere l'identificazione dell'appartenenza di una qualsiasi data, correttamente inserita in formato gg/mm, alla corrispondente stagione

↙ mese · 100 + giorno

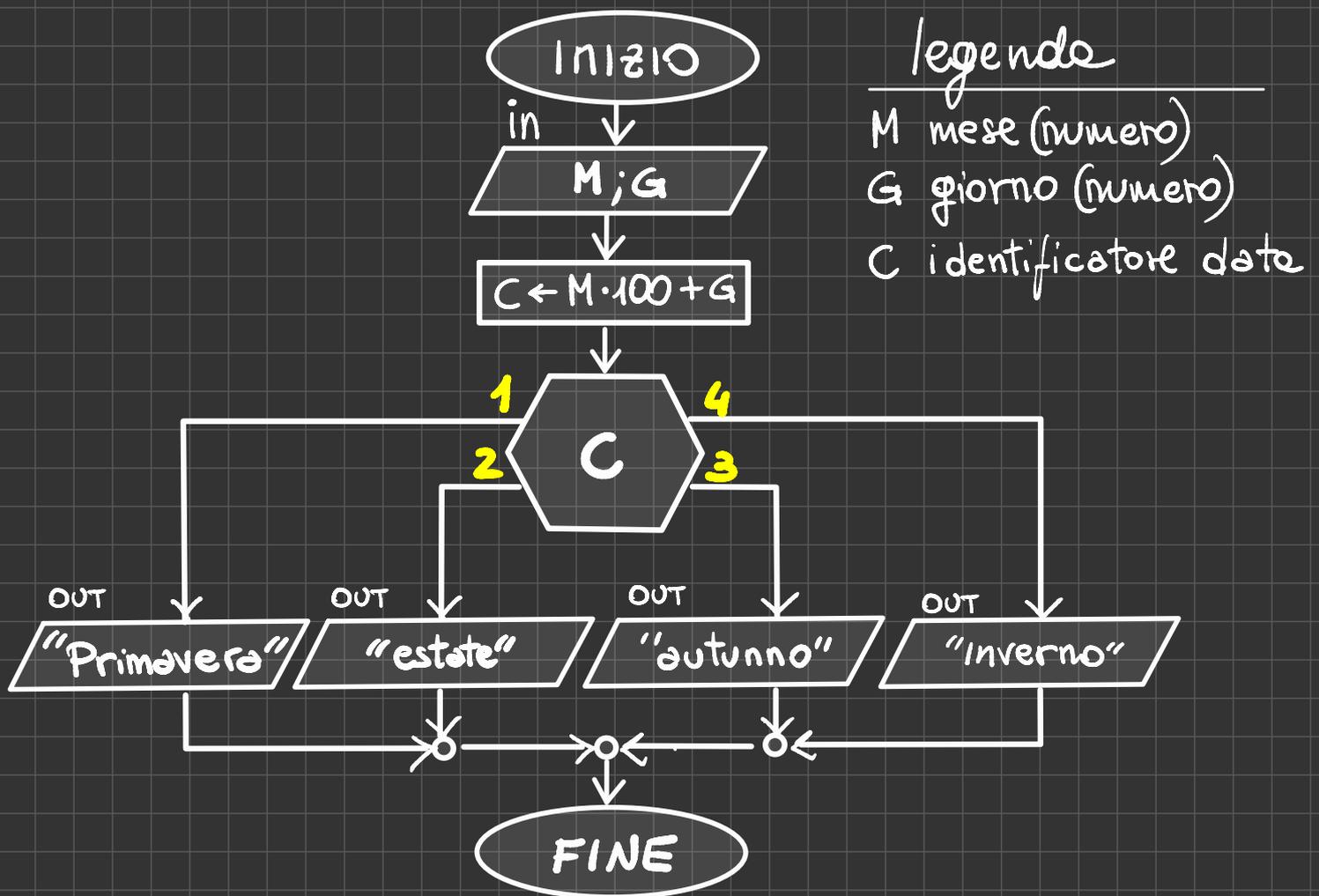
giorno	mese	C	
01	01	0101	Inverno
⋮			
20	03	0320	Primavera
21	03	0321	
⋮			
20	06	0620	estate
21	06	0621	
⋮			
22	09	0922	autunno
23	09	0923	
⋮			
20	12	1220	Inverno
21	12	1221	
⋮			
31	12	1231	



d'algoritmo risulta ben ingegnerizzato, rispetta le specifiche sulle topologie e sulle prestazioni, ma possiamo ancora migliorare la topologie introducendo il blocco "case"

Il blocco "case" (switch - case) rende più snello il diagramma, potendo raggruppare tutti i controlli!

Il blocco "case" viene utilizzato quando è necessario eseguire controlli multipli sulle stesse variabili



case 1)  $320 < C < 621$  →  $(C > 320) \text{ AND } (C < 621)$

case 2)  $620 < C < 923$  →  $(C > 620) \text{ AND } (C < 923)$

case 3)  $922 < C < 1221$  →  $(C > 922) \text{ AND } (C < 1221)$

case 4)  $C < 321$   
 $C > 1220$  →  $(C < 321) \text{ OR } (C > 1220)$