

Algoritmo di Euclide per il Calcolo del Massimo Comun Divisore

Dati 2 numeri interi positivi a, b calcolare il massimo comun divisore di a, b

Dati di input

a, b : coppia di numeri interi > 0

Dati di output

mcd : numero intero > 0

L'algoritmo si basa sulle seguenti proprietà:

$$Mcd(a, b) = a = b \quad \text{se } a = b \quad (1.)$$

$$Mcd(a, b) = Mcd(a - b, b) \quad \text{se } a > b \quad (2.)$$

$$Mcd(a, b) = Mcd(a, b - a) \quad \text{se } a < b \quad (3.)$$

Considerazioni

La proprietà (1.) afferma che il massimo comun divisore di (a, b) è proprio a (e/o b).

Le proprietà (2.) e (3.) affermano che il calcolo del massimo comun divisore di (a, b) può ricondursi al calcolo del massimo comun divisore di $(a - b, b)$ se $a > b$ oppure al calcolo del massimo comun divisore di $(a, b - a)$ se $a < b$.

Idea dell'algoritmo

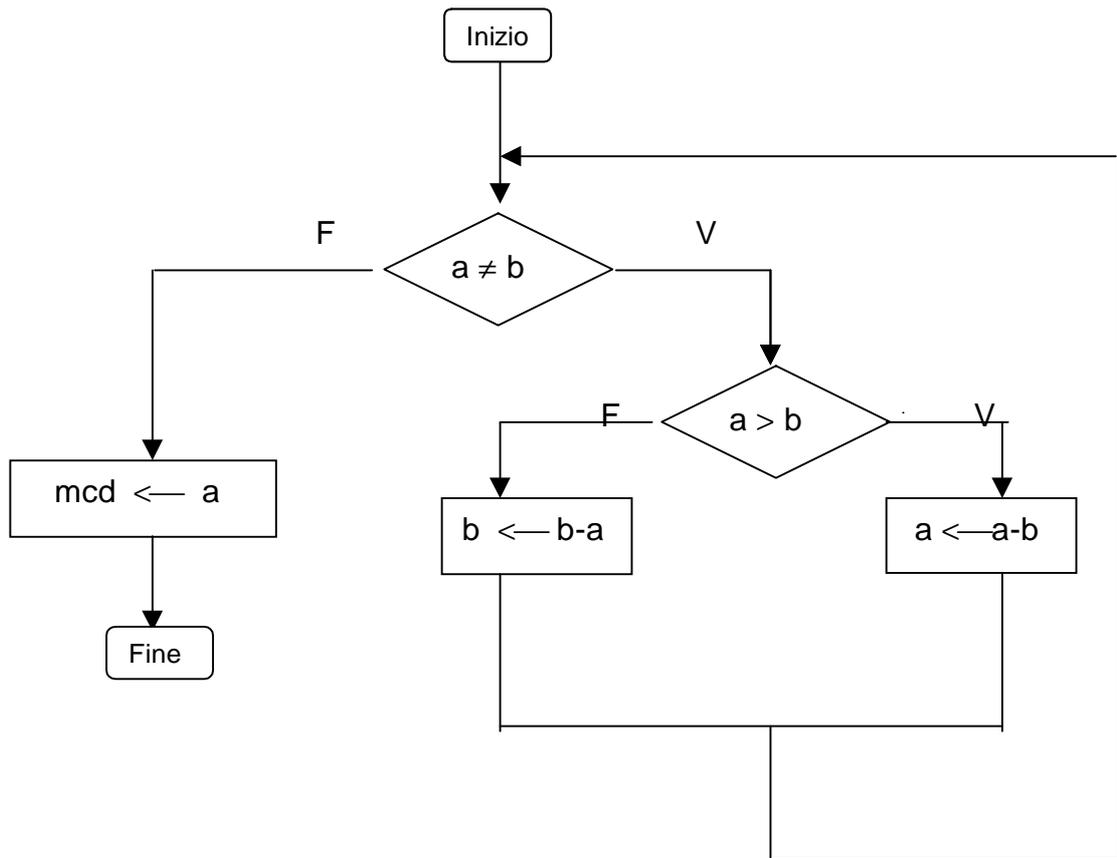
Poiché nel caso in cui $a = b$ (1.) il problema è risolto, ed inoltre valgono le proprietà (2.) e (3.) nel caso generale si possono effettuare tante differenze in modo da riportarsi al caso $a = b$.

In particolare è necessario sottrarre dal $\max(a, b)$ l'altro elemento della coppia di numeri: tale operazione deve essere ripetuta fino a quando non si abbia $a = b$, la sottrazione cioè dia risultato uguale a zero.

Se si sceglie di esprimere tale soluzione con una iterazione a condizione iniziale (*While...do.*), la condizione è $(a \neq b)$.

Algoritmo (*pseudocodifica in Pascal*)

```
While (a≠b) do
  Begin
    If (a>b)
      Then assegna ad a il valore a-b
      Else assegna a b il valore b-a
    End
  Assegna a mcd il valore a
```



Codifica in Pascal

```
procedure mcd_euclide (a,b:integer; var mcd: integer);  
  (* La procedura calcola il massimo comun divisore  
  di a, b mediante l'algorithmo di Euclide *)  
  Begin  
    while ( a<>b) do  
      if ( a>b ) then a:=a-b  
      else b:=b-a;  
    mcd:=a  
  End;
```