

RICHIAMI IMPORTANTI DI PROGRAMMAZIONE (esempi dal manuale della calcolatrice)

Esecuzione di un programma

Dallo schermo base:

```
prog1()  
└─ Se gli argomenti non sono  
  richiesti
```

```
prog1(x,y)  
└─ Se gli argomenti sono  
  richiesti
```

Comando If

Per eseguire un solo comando se un test condizionale è vero, utilizzare la seguente formula generale.

```
:If x>5  
❶ : Disp "x is greater than 5"  
❷ :Disp x
```

- ❶ Eseguito solo se $x > 5$; in caso contrario viene saltato.
- ❷ Visualizza sempre il valore di x .

Strutture If...Then...EndIf

Per eseguire comandi multipli se un test condizionale è vero, utilizzare la seguente struttura:

```
:If x>5 Then  
❶ : Disp "x is greater than 5"  
❶ : 2*x→x  
❷ :EndIf  
:Disp x
```

- ❶ Eseguito solo se $x > 5$.
- ❷ Visualizza il valore di:
 - $2x$ if $x > 5$.
 - x if $x \leq 5$.

Nota: EndIf contrassegna la fine del blocco Then eseguito se la condizione è vera.

Strutture If...Then...Else... EndIf

Per eseguire un gruppo di comandi se un test condizionale è vero e, viceversa, per eseguire un altro gruppo di comandi se la condizione è falsa, utilizzare la seguente struttura.

```
  :If x>5 Then
❶ : Disp "x is greater than 5"
❶ : 2*x→x
  :Else
❷ : Disp "x is less than or
❷   equal to 5"
  : 5*x→x
  :EndIf
❸ :Disp x
```

- ❶ Eseguito solo se $x > 5$.
- ❷ Eseguito solo se $x \leq 5$.
- ❸ Visualizza il valore di:
 - $2x$ if $x > 5$.
 - $5x$ if $x \leq 5$.

Strutture If...Then...Elseif... EndIf

```
:g(x)
:Func
:If x≤-5 Then
: Return 5
: ElseIf x>-5 and x<0 Then
: Return -x
: ElseIf x≥0 and x≠10 Then
: Return x
: ElseIf x=10 Then
: Return 3
:EndIf
:EndFunc
```

Comandi Lbl e Goto

```
  :If x>5
❶ : Goto GT5
❷ :Disp x
  :-----
  :-----
  :Lbl GT5
  :Disp "The number was > 5"
```

- ❶ Se $x > 5$, passa direttamente all'etichetta GT5.
- ❷ In questo esempio, il programma deve includere comandi (come **Stop**) che impediscano l'esecuzione di Lbl GT5 se $x >$ oppure = 5.

Cicli For...EndFor

```
      :For i, 0, 5, 1  
❶ : Disp I  
      :EndFor  
❷ :Disp i
```

- ❶ Visualizza 0, 1, 2, 3, 4 e 5.
- ❷ Visualizza 6. Quando il valore di *variabile* raggiunge 6, il ciclo non viene eseguito.

Nota: è possibile definire la variabile del contatore come locale se non occorre salvarla dopo l'interruzione del programma.

Cicli While...EndWhile

```
❶ :0→x  
   :While x<5  
❷ : Disp x  
❸ : x+1→x  
   :EndWhile  
❹ :Disp x
```

- ❶ Impostazione iniziale di x.
- ❷ Visualizza 0, 1, 2, 3 e 4.
- ❸ Incrementi di x.
- ❹ Visualizza 5. Quando x raggiunge il valore di 5, il ciclo non viene eseguito.

Cicli Loop...EndLoop

```
      :0→x  
      :Loop  
      : Disp x  
      : x+1→x  
❶ : If x>5  
      : Exit  
      :EndLoop  
❷ :Disp x
```

- ❶ Un comando **If** controlla la condizione.
- ❷ Esce dal ciclo e passa a questa posizione quando x raggiunge 6.

Nota: il comando **Exit** permette di uscire dal ciclo corrente.

In questo esempio, il comando **If** può essere in qualsiasi punto del ciclo.