

Come potete vedere, nei casi esposti sopra non si conosce dapprima il necessario numero di ripetizioni, vediamo cosa accade se invece lo si conosce dal principio. Per questi casi si utilizza l'istruzione **for...next**. La sintassi di For... Next è la seguente:

**For Contatore = Inizio To Fine [ Step Incremento ]**

**[Bloccolstruzioni]**

**[Exit For]**

**[Bloccolstruzioni]**

**Next [Contatore]**

dove:

- **Inizio To Fine** specificano il valore iniziale e finale che assumerà **Contatore**, procedendo di Incremento in Incremento;
- se **Incremento** è omesso, vale 1; se invece è negativo, allora il ciclo va a ritroso, e Inizio deve essere maggiore di Fine;
- **Exit For** permette di uscire da un ciclo a metà (per esempio perché ho trovato in anticipo ciò che stato cercando, e non voglio perdere tempo ad arrivare fino alla fine).

## Ripetizione PRE-Condizionale

Gli appunti di [www.scuola24ore.net](http://www.scuola24ore.net) prof.ssa Patrizia Tarantino

### LA RIPETIZIONE PRE\_ CONDIZIONALE

**Do Until** <condizione> a differenza del *do loop*, *Until* viene analizzata prima la condizione, quindi verrà effettuata la prima ripetizione solo se si è verificata la condizione  
**Loop** <istruzioni1> verranno eseguite solo se la condizione è vera.

```

        Ripetizione PRECondizionale per VERO.vb
        Ripetizione PRECond...0.vb (Progettazione)
        (Generale)
        Public Class Form1
        Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As EventArgs) Handles MyBase.Load
            Dim materia, risposta As String
            Dim voto, media, somma As Decimal
            Dim cont As Integer
            risposta = InputBox("Vuoi continuare ? SI/NO")
            Do Until risposta = "no"
                materia = InputBox("Inserisci materia")
                voto = InputBox("Inserisci voto")
                somma = somma + voto
                cont = cont + 1
                risposta = InputBox("Vuoi continuare ? SI/NO")
            Loop
            media = somma / cont
            MsgBox("La Media e..." & media)
        End Sub
        End Class
    
```

*struttura DERIVATA*

**Do While** <condizione> Come per la struttura del *Do...Loop while* la condizione verrà analizzata così *finché NON* condizione...ripeti  
**Loop** <istruzioni> verranno eseguite solo se la condizione non è vera.

```

        Ripetizione PRE Condizionale per FALSO.vb
        Ripetizione PRE Cond...0.vb (Progettazione)
        (Generale)
        Public Class Form1
        Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As EventArgs) Handles MyBase.Load
            Dim materia, risposta As String
            Dim voto, media, somma As Decimal
            Dim cont As Integer
            risposta = InputBox("Elenco finito - SI/NO")
            Do While risposta = "no"
                materia = InputBox("Inserisci materia")
                voto = InputBox("Inserisci voto")
                somma = somma + voto
                cont = cont + 1
                risposta = InputBox("Elenco finito - SI/NO")
            Loop
            media = somma / cont
            MsgBox("La Media e..." & media)
        End Sub
        End Class
    
```

*MENTRE RIPETI .....*  
 oppure  
*FINCHE' NON CONDIZIONE RIPETI .....*

*struttura DERIVATA*

## Ripetizione POST-Condizionale

gli appunti di www.scuola24ore.net

prof.ssa Patrizia Tarantino

### LA RIPETIZIONE POST\_ CONDIZIONALE

**Do**

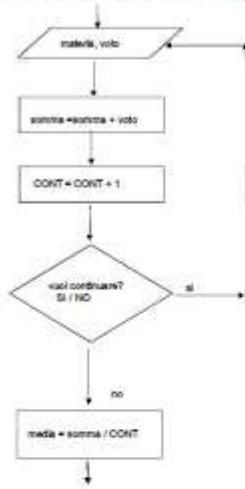
<istruzioni> → vengono eseguite sempre almeno una volta

**Loop Until**

<condizione> → espressione tipo  $A=0; B>500; NOME = 'MARIA'$

<istruzioni>

verranno ripetute solo se la condizione è vera.



### RIPETI FINCHE'...

```

RipetizionePOST CondizionalePerVERO.vb  RipetizionePOST Cond...
[Event1 Form1]
Public Class Form1
    Private Sub Form1_Load(ByVal sender As Sys
        Dim materia, risposta As String
        Dim voto, media, somma As Decimal
        Dim cont As Integer

        Do
            materia = InputBox("Inserisci mate
            voto = InputBox("Inserisci voto")
            somma = somma + voto
            cont = cont + 1
            risposta = InputBox("Vuoi continua
        Loop Until risposta = "no"
        Media = somma / cont
        MsgBox("La Media è..." & media)
    End Sub
End Class
    
```

struttura DERIVATA

**Do**

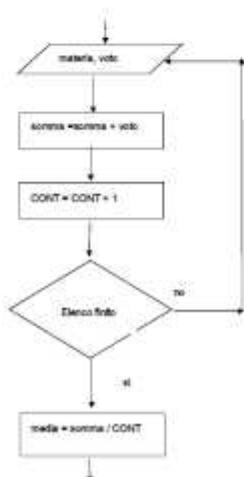
<istruzioni> → vengono eseguite sempre almeno una volta

**Loop While**

<condizione> → espressione tipo  $A=0; B>500; NOME = 'MARIA'$

<istruzioni>

verranno eseguite solo se la condizione non è vera.



### RIPETI MENTRE...

```

RipetizionePOSTcondizionale per FALSO.vb  RipetizionePOSTcond...
[Event1 Form1]
Public Class Form1
    Private Sub Form1_Load(ByVal sender As System
        Dim materia, risposta As String
        Dim voto, media, somma As Decimal
        Dim cont As Integer

        Do
            materia = InputBox("Inserisci materia"
            voto = InputBox("Inserisci voto")
            somma = somma + voto
            cont = cont + 1
            risposta = InputBox("Finisce finito - F
        Loop While risposta = "no"
        Media = somma / cont
        MsgBox("La Media è..." & media)
    End Sub
End Class
    
```

## Le ripetizioni

Abbiamo finora conosciuto 2 strutture di controllo: **la struttura sequenziale**, e **la struttura alternativa**. La terza struttura fondamentale della programmazione strutturata è la **ripetizione (CICLO)**. Per ciclo si intende un meccanismo per il quale un blocco di istruzioni viene eseguito più di una volta, e più precisamente:

- un numero di volte noto a priori: istruzione **For... Next**
- un numero di volte non noto a priori: istruzione **Do... Loop Until**

Nel secondo caso è evidente che occorre prevedere una condizione (sempre del tipo che dia come risultato Vero o Falso) per permettere al ciclo di fermarsi. In caso contrario il ciclo continuerebbe all'infinito. I cicli sono fondamentali, in quanto esistono moltissime operazioni, che tipicamente fa un'applicazione, che richiedono di essere eseguite più volte. Le istruzioni di controllo eseguite una sola volta (che cioè non generano dei cicli) sono rappresentate dall'istruzione If... Then ... Else che abbiamo già visto nelle lezioni precedenti e dall'istruzione Select Case, che vi illustrerò in una lezione più avanti. Qui sotto vi mostro gli algoritmi che mettono a confronto la ripetizione post-condizionale e pre-condizionale con le relative traduzioni in Visual Basic. Cliccate le immagini per leggere i due file riassuntivi in formato Pdf (stampabile), oppure cliccate qui per provare le stesse istruzioni nel progetto Visual basic: Progetto [RIPETIZIONI](#)