

ESTRUCTURAS DE CONTROL.

Una estructura de control permite controlar el flujo de la ejecución de instrucciones. Con estas estructuras, el programador puede determinar el orden en que se ejecutarán las instrucciones que están dentro de estas estructuras. Teorema de programación: “Todo programa propio se puede escribir utilizando únicamente las estructuras de control secuencial, condicional e iterativa” Un programa propio es aquel que:

- ♣ Tiene un único punto de entrada y un único punto de salida.
- ♣ Existen caminos desde la entrada hasta la salida que pasan por todas las partes del programa.
- ♣ Todas las instrucciones son ejecutables y no existen bucles sin fin. Las estructuras conocidas son:
 - ♣ Secuencial: Se caracterizan porque una acción se ejecuta detrás de otra. El flujo del programa coincide con el orden físico en el que se ha ido poniendo las instrucciones.



- ♣ Selectiva simple: Se evalúa la condición y si esta da como resultado verdad se ejecuta una determinada acción o grupo de acciones, en caso contrario se salta dicho grupo de acciones.



SI <condición> **entonces**
 <Acción o acciones>
fin_si

- ♣ Selectiva doble: Cuando el resultado de evaluar la condición es verdad se ejecuta una determinada acción o grupo de acciones y si el resultado es falso otra acción o grupo de acciones diferentes.

CONDICION	
SI	NO
ACCION 1	ACCION 2

SI <condición> **entonces**
 <Acción 1>
SI_NO
 <Acción 2>
fin_si

♣ Selectiva múltiple: Se ejecutan unas acciones u otras según el resultado que se obtenga al evaluar una expresión.

CONDICION			
V1	V2	V3	Vn
ACCION 1	ACCION 2	ACCION 3	ACCION N

♣ Repetitivas: Las estructuras que repiten una secuencia de instrucciones un número determinado de veces o denominadas bucles, y se llama iteración al hecho de repetir la ejecución de una secuencia de acciones.

MIENTRAS	CONDICION
ACCIONES A REPETIR	

If – Else.

Sentencia If

Representa una toma de decisión sencilla, eso quiere decir, que si la condición que se evalúa es verdadera se realizan las instrucciones que pertenezcan al bloque If y continuará con el resto del programa.

```
5      'Declaro y asigno valor a una variable para poder utilizarla como condición
6      Dim variable As String = "Hola"
7
8      'La condición es si la variable tiene como valor "Hola".
9      If (variable = "Hola") Then
10         'Instruccion/es a ejecutarse si la condición es verdadera.
11         'En este ejemplo lo sería.
12     End If
```

Se pueden anidar varios If, es decir, uno entre otro, para ingresar cada vez al If más interno es necesario que la condición sea verdadera.

Sentencia If – Else

Es una estructura compuesta que evalúa una condición. Si esta es verdadera realizará las instrucciones contenidas en el bloque If, en caso contrario ingresará por el bloque Else.

```
4
5      'Declaro y asigno valor a una variable
6      Dim usuario As String = "pepito"
7
8      'La condición es verdadera (se cumple) si la variable tiene como valor "pepe"
9      If (usuario = "pepe") Then
10         'Instruccion/es a ejecutarse si la condición es verdadera.
11     Else
12         'Instruccion/es a ejecutarse si la condición NO es verdadera.
13     End If
```

Select

Es una toma de decisión con varias opciones, o sea, según el valor que tome la variable a evaluar se escogerá un caso entre varias alternativas. También cuenta con un caso por defecto en caso de que la variable no tome ninguno de los valores considerados.

```

17 Dim num As Integer = 13
18 Select Case num
19     Case 2
20         'Instruccion/es a ejecutarse si la variable num toma valor 2
21     Case 8
22         'Instruccion/es a ejecutarse si la variable num toma valor 8
23     Case Else
24         'Instruccion/es a ejecutarse si la variable num no toma ningun valor anterior
25 End Select

```

For – Next.

Se utiliza para repetir una o varias instrucciones, usando una variable que por lo general es llamada contador.

```

8 Dim valor As Integer
9 For valor = 1 To 5 Step 1
10     MsgBox(valor)
11 Next

```

Consta de 3 partes: inicialización, condición y actualización (opcional). El valor de la variable es inicializado en valor = 1, to 5 representa el valor final del ciclo repetitivo. La sentencia Step es opcional, por defecto sino se utiliza el incremento será de 1.

While

Se utiliza para repetir una secuencia de instrucciones siempre que la condición evaluada sea verdadera. Si al evaluar la condición la misma es falsa no ingresará al ciclo y continuará con el resto del programa.

```

7
8 'Valor inicial de la variable
9 While (condición )
10     'Instruccion/es
11     'Incremento o decremento
12 End While

```

Do – Loop

Permite ejecutar por lo menos una vez al grupo de instrucciones. Al contrario de While, Do-Loop primero ejecuta una vez el ciclo y luego evalúa si la condición se cumple, en el caso que no se cumpla dejará de ejecutarse.

```
7  
8      Do  
9          'Instrucción/es  
10         'Incremento o decremento  
11     Loop While ~(condición)  
12
```

For – Each

Sirve para repetir uno a varias instrucciones por cada elemento de una colección. Se utiliza cuando en una colección no se tiene conocimiento de cuantos elementos existen.

```
14  
15     For Each 'elemento [As tipo_de_dato] In colección  
16         'Instrucción/es  
17     Next
```