

INTRODUCCIÓN A LA PROGRAMACIÓN

UNIDAD 3 – PSEUDOCÓDIGO

UNIDAD: 3

PRESENTACIÓN: Bucles anidados



Bucles Anidados

Como parte del bloque de sentencias que pertenece al cuerpo de un bucle, se pueden incluir otros bucles, y es lo que se llama anidamiento de bucles. Esto proporciona una mayor flexibilidad a la hora de resolver ciertos problemas. Comencemos con el caso más sencillo de anidamiento de bucles, el de dos bucles anidados, es decir, el caso de un bucle dentro de otro.

El bucle interno (b.i.) es una sentencia más, dentro del bloque de sentencias del bucle externo (b.e.), por lo tanto, el ciclo completo del b.i. se ejecuta repetidamente tantas veces como veces se repita el b.e.

Esto quiere decir que una sentencia dentro del cuerpo del b.i. se va a ejecutar, las “n” veces que se repetirían en el caso de que no fuera anidado el b.i., por las “m” veces que se ejecutan cada una de las sentencias del b.e., es decir, “m * n” veces. Paso a paso, sucedería del siguiente modo:

1. Se evalúa la expresión condicional del bucle más externo. Si es falsa, se sale de los bucles y continúa la ejecución del resto del programa. Si es verdadera continúa con el paso 2.
2. Se ejecuta el bloque de sentencias del bucle más externo. Entre ellas, se encontrará el bucle mas interno.
 - 2.1. Se evalúa la expresión del bucle mas interno. Si es falsa, sale del bucle interno y continúa con la ejecución del bucle externo, en el paso 3. Si es verdadera continúa en el paso 2.2.
 - 2.2. Se ejecutan el bloque de sentencias del bucle interno.
 - 2.3. Vuelve al paso 2.1.
3. Vuelve al paso 1.

Ejemplo:

```
#include <stdio.h>
main()
{short i, j;

    printf("La tabla de multiplicar del 1 al 10 es:\n");
    for (i=1;i<=10;i++)

        {for ( j = 1; j <= 10; j++ )
        printf("%d por %d vale %d",i,j,i*j);
        }

    return 0;}
```

Se pueden anidar muchos niveles y con diferentes tipos de bucles.