



APRENDERAPROGRAMAR.COM

BUCLES EN LENGUAJE C
(ESTRUCTURA DE
REPETICIÓN) CONDICIÓN,
CONTADOR. EJEMPLOS.
(CU00533F)

Sección: Cursos

Categoría: Curso básico de programación en lenguaje C desde cero

Fecha revisión: 2031

Resumen: Entrega nº33 del curso básico "Programación C desde cero".

Autor: Mario Rodríguez Rancel

BUCLES FOR EN LENGUAJE C

La instrucción `for` explota la capacidad del ordenador para repetir procesos y para contar. Muchas veces esta instrucción se puede reemplazar por otras instrucciones o estructuras de tipo repetición. Se podrá optar por la que se estime más oportuna, que muchas veces será la instrucción `for` pues reúne una cualidades interesantes.



La sintaxis tipo a emplear con C es la siguiente:

```
int VAR;
for (VAR = Vi; VAR<Vf; VAR++ ) {
    Instrucción 1;
    Instrucción 2;
    .
    .
    .
    Instrucción n;
}
```

El primer término dentro del paréntesis del `for` inicializa `VAR` a un valor inicial `Vi`. `VAR` hace referencia a la variable que sirve como guía del bucle. Normalmente se usan nombres de variable como `i`, `j`, `k`... El segundo término indica la condición de finalización del bucle, para lo cual se puede usar una condición como `VAR < Vf`, `VAR <= Vf`, `VAR > Vf` ó `VAR >= Vf`. El tercer término indica el incremento que sufre el contador en cada pasada del bucle. La sintaxis `VAR++` indica que el incremento será de una unidad. Por el contrario `VAR--` indica que habrá un decremento en una unidad. Si usamos `VAR++` el primer valor que toma `VAR` en el bucle será `Vi`, el segundo `Vi + 1`, el tercero `(Vi + 1) + 1`, etc. La modificación de dicho valor de incremento se realiza en C escribiendo una condición del tipo `VAR = VAR + numero`. Por ejemplo `i = i + 5` en un contador de un bucle `for` indicará que en cada repetición del bucle el contador se incrementará en 5 unidades. Un paso negativo del tipo `i--` supone que el contador decrece en vez de incrementarse. Si el paso es negativo, `Vi` necesariamente habrá de ser mayor que `Vf`, ya que en caso contrario no se producirá la entrada en el bucle.

Con este código se nos muestra en pantalla 3 veces *hola*, gracias a que se ha invertido el sentido del bucle:

```
#include <stdio.h>
#include <stdlib.h>
int main() {
    int VAR, Vi, Vf; Vi=1; Vf=3;
    for (VAR=Vf; VAR>=Vi; VAR--) { puts("hola\n"); }
    return 0; // Ejemplo aprenderaprogramar.com
}
```

Con C de estándares anteriores era posible usar sintaxis del tipo `for (int i=0; i<10; i++) { ... }` donde la variable del bucle se declaraba dentro del propio bucle. En el último estándar de C esto no se permite y en caso de emplearse se obtiene un error similar a: "Sólo se permiten las declaraciones iniciales del bucle 'for' en modo C99. Use la opción `-std=c99` o `-std=gnu99` para compilar su código".

Resulta admisible usar `for` sin llaves para encerrar las instrucciones afectadas por el bucle, por ejemplo `for (VAR=VF; VAR>=VI; VAR--) puts("hola\n");` En este caso el bucle comprende únicamente la primera instrucción a continuación del `for`. Sin embargo, no recomendamos usar el `for` sin llaves pues puede dificultar la lectura y depuración de los programas.

La instrucción `for` es anidable dentro de sí misma o dentro de otros tipos de bucles o estructuras, debiendo seguir las pautas que ya hemos comentado.

Recomendamos leer lo indicado en el curso "Bases de la programación nivel I" de aprenderaprogramar.com, apartado correspondiente al "Uso y mal uso de la instrucción Desde ... Siguiente" para evitar problemas con la sentencia `for` de C.

EJERCICIO RESUELTO N°1: ENUNCIADO

Transformar en código el pseudocódigo que se muestra a continuación, relativo a un programa que pide un número del 1 al 10 y nos muestra la tabla de multiplicar correspondiente.

1. Inicio [Pseudocódigo aprenderaprogramar.com]
2. Mostrar "¿Qué tabla quiere conocer (de 1 a 10)?"
3. Pedir T
4. Mostrar "Tabla del", T
5. Desde i = 1 hasta 10 Hacer
 - Mostrar T, "*", i, "=", T * i
- Siguiente
6. Fin

EJERCICIO RESUELTO N°1: SOLUCIÓN

En este programa usamos la variable T para almacenar el número introducido por el usuario, y la variable i como contador en el bucle. La sentencia `puts` con una cadena vacía como argumento la usamos para introducir un salto de línea.

```
#include <stdio.h>
#include <stdlib.h>
int main() {
    int T, i;
    printf("Indique la tabla que desea conocer (de 1 a 10): ");
    scanf("%d", &T);
    puts(""); printf("TABLA DEL %d \n", T);
    for (i=1; i<=10; i++) { printf("%d * %d = %d \n", T, i, T*i); }
    return 0; // Ejercicios resueltos aprenderaprogramar.com
}
```

Un resultado de ejecución podría ser el siguiente:

Indique tabla que desea conocer (de 1 a 10): 8

TABLA DEL 8

8 * 1 = 8

8 * 2 = 16

8 * 3 = 24

8 * 4 = 32

8 * 5 = 40

8 * 6 = 48

8 * 7 = 56

8 * 8 = 64

8 * 9 = 72

8 * 10 = 80

EJERCICIO RESUELTO N°2: ENUNCIADO

Transformar en código el pseudocódigo que se muestra a continuación, relativo a un programa que muestre todos los enteros comprendidos entre un número definido por el usuario y 100. Razonar la lógica del programa y valorar si es correcta o no.

1. Inicio [Pseudocódigo aprenderaprogramar.com]
2. Mostrar "Introduzca un número entero comprendido entre 1 y 99" : Pedir num
3. Desde $i = \text{num}$ hasta 100 Hacer
Mostrar i
Siguiente
4. Fin

EJERCICIO RESUELTO N°2: SOLUCIÓN

```
#include <stdio.h>
#include <stdlib.h>
int main() {
    int num, i;
    printf("Introduzca un numero entero comprendido entre 1 y 99: ");
    scanf("%d", &num);
    puts(""); printf("ENTEROS ENTRE %d y %d \n", num, 100);
    for (i=num; i<=100; i++) { printf("%d \t", i); }
    return 0; // Ejercicios resueltos aprenderaprogramar.com
}
```

Un resultado de ejecución podría ser este:

```
Introduzca un numero entero comprendido entre 1 y 99: 86
ENTEROS ENTRE 86 y 100
86  87  88  89  90  91  92  93  94  95 96
97  98  99  100
```

EJERCICIO Nº 1

Escribe un programa que use un bucle for para mostrar el resultado de multiplicar los números del 1 al 20 por el resultado de obtener el módulo (resto de la división que se obtiene usando el operador % de C) de dicho número con un número elegido por el usuario.

Ejemplo: el usuario elige el número tres. Los cálculos a realizar serán: resto de 1 entre 3 es 1, por pantalla se mostrará un 1, resto de 2 entre 3 es 2, por pantalla se mostrará un 2, resto de 3 entre 3 es cero, por pantalla se mostrará un cero, resto de 4 entre 3 es 1, por pantalla se mostrará un 1... así hasta terminar con resto de 20 entre 3 es 2, por pantalla se mostrará un 2.

Para comprobar si tus respuestas son correctas puedes consultar en los foros aprenderaprogramar.com.

EJERCICIO Nº 2

Escribe un programa que use un bucle for para calcular la potencia de un número al que denominaremos base sobre otro número al que denominaremos exponente, ambos elegidos por el usuario.

Ejemplo: el usuario elige como base el 4 y como exponente el 3. El programa debe mostrar el resultado de multiplicar la base por sí misma 3 veces, en este ejemplo $4*4*4 = 64$.

Para comprobar si tus respuestas son correctas puedes consultar en los foros aprenderaprogramar.com.

Próxima entrega: CU00534F

Acceso al curso completo en [aprenderaprogramar.com](http://www.aprenderaprogramar.com) --> Cursos, o en la dirección siguiente:
http://www.aprenderaprogramar.com/index.php?option=com_content&view=category&id=82&Itemid=210