

Estructuras repetitivas

Favian Orduña Suárez

Tecnológico Nacional de México Campus Querétaro

Fundamentos de programación

María Luisa Montes Almanza

08 de octubre del 2025

¿Que son las estructuras repetitivas?

Son todas aquellas aplicaciones en las que una operación o conjunto de estas se deben de repetirse mas de una vez. La repetición de una acción o varias se llevan a cabo mientras se cumpla una cierta condición. Dicha condición puede estar en el ciclo desde (for); los bucles mientras (while) y hacer-mientras (do_while) (Osvaldo cairó, 2005; Corona Nakamura, 2011).

Estos ciclos o bucles son un segmento de algoritmo, donde las instrucciones se repiten un numero conocido o indefinido de veces mientras se cumpla una condición determinada. Cada vuelta se vuelve a realizar la condición y si esta es verdadera entonces se repite el bucle hasta que esta condición sea falsa para dar fin al bucle (Osvaldo cairó, 2005; Corona Nakamura, 2011).

Estructura FOR

Esta es la mas utilizada y simple porque se repite un conjunto de instrucciones un numero determinada de veces. Se utiliza un contador para poder llevar la cuenta de veces que se ha repetido el bucle para que cuando el contador llegue al termino que hace que ya no cumpla la condición entonces pare y salga del bucle dando por terminado esta estructura (Osvaldo cairó, 2005; Corona Nakamura, 2011).

Figura 1: Maneras de poner la estructura FOR

Diagrama de flujo	Pseudocódigo	Lenguaje C
	<p>Con una instrucción</p> <p>desde (expr_ini(s),cond,inc(s))</p> <p>Inst 1</p>	<p>Con una instrucción</p> <p>for (expr_ini (s) ; cond; inc (s))</p> <p>Inst 1</p>
	<p>Con varias instrucciones</p> <p>desde (expr_ini(s),cond,inc(s))</p> <p>inicio</p> <p>Inst 1</p> <p>Inst 2</p> <p>—</p> <p>Inst n</p> <p>fin</p>	<p>Con varias instrucciones</p> <p>for (expr_ini (s) ; cond; inc (s))</p> <p>{</p> <p>Inst 1</p> <p>Inst 2</p> <p>—</p> <p>Inst n</p> <p>}</p>

Fuente: Diseño de algoritmos y su codificación en c (2011)

Estructura WHILE

Cuando no sabes cuantas veces debería repetirse un bucle pero si hasta que punto entonces se utiliza el while el cual va a estar repitiendo una serie de instrucciones mientras se cumpla una condición. Cuando la condición se cumple entonces da como valor de 1 que es verdadero entonces se van a realizar las instrucciones pero si no se cumple la condición entonces da un valor de 0 que significa falso lo cual indica que ya no se realizaran las instrucciones del bucle y seguirá con la secuencia normal (Osvaldo cairó, 2005; Corona Nakamura, 2011).

Figura 2: Maneras de poner la estructura WHILE

Diagrama de flujo	Pseudocódigo	Lenguaje C
	Con una instrucción	Con una instrucción
	mientras (condición)	while (condición)
	Inst 1	Inst 1
	Con varias instrucciones	Con varias instrucciones
	mientras (condición)	while (condición)
	inicio	{
	Inst 1	Inst 1
	Inst 2	Inst 2
	-	-
	Inst n	Inst n
	fin	}

Fuente: Diseño de algoritmos y su codificación en c (2011)

Estructura DO_WHILE

Al igual que la estructura while este se utiliza cuando no se sabe cuantas veces se quiere repetir el bucle pero a diferencia del otro este por lo menos deja realizar 1 vez las instrucciones que están dentro del bucle. Esto pasa por que la condición la aplica después de realizar el bucle y el while lo hace antes de realizar el bucle. Entonces si se cumple cierta condición entonces se va a volver a repetir desde cierta parte del algoritmo y tendría que llegar otra vez a la condición en donde va a poder continuar hasta que la condición sea falsa para que pueda salir del bucle y continué con la secuencia lineal de lo contrario se volvería a repetir la secuencia de instrucciones (Osvaldo cairó, 2005; Corona Nakamura, 2011).

Figura 3: Maneras de poner la estructura DO_WHILE

Diagrama de flujo	Pseudocódigo	Lenguaje C
	Con una instrucción	Con una instrucción
	hacer Inst 1 mientras (condición)	do Inst 1 while (condición)
	Con varias instrucciones	Con varias instrucciones
	hacer inicio Inst 1 Inst 2 — Inst n fin mientras (condición)	do { Inst 1 Inst 2 — Inst n } while (condición);*

Fuente: Diseño de algoritmos y su codificación en c (2011)

Referencias

- Osvaldo Cairó (2005). Metodología de la programación Algoritmos, diagramas de flujo programas (3a ed.). Editorial Alfaomega
- Corona Nakamura, María de los ángeles (2011). Diseño de algoritmos y su codificación en lenguaje c (1ª ed.). Editorial Mc Graw Hill