Nombre: Jose Javier Bonilla Salazar
Carnet: 202200035
Clase: Introducción a la programación
Profesor: Brayan Prado Marroquín

Investigación Hoja de trabajo 3

Contenido

Introducción

Los distintos tipos de operadores tanto lógicos como aritméticos son de gran ayuda en el ámbito de la programación, ya que gracias a ellos se logra realizar operaciones bastante complejas que permiten realizar programas con una serie de instrucciones complejas.

1. Marco teórico

1.1. Operaciones relacionales

Los operadores relacionales comparan datos numéricos, de serie de caracteres o lógicos. El resultado de la comparación ya sea Verdadero (1) o falso (0), puede utilizarse para tomar una decisión referente al flujo del programa.

(Operadores relacionales, s. f.)

Figura 1. **Tipos de operadores relacionales**

Operador	Relación	Ejemplo
EQ o =	Igualdad	X = Y
NE o #	Desigualdad	X # Y
>< o <>	Desigualdad	X <> Y
LT o <	Menor que	X < Y
GT o >	Mayor que	X > Y
LE o <= o =< o #>	Menor que o igual a	X <= Y
GE o >= o => o #<	Mayor que o igual a	X >= Y

Fuente: https://www.ibm.com/docs/es/iis/11.5?topic=bo-relational-operators\

Cuando en una expresión se utilizan tanto operadores aritméticos como operadores relacionales, las operaciones aritméticas se realizan primero.

(*Operadores y expresiones*, s. f.)

1.2. Operaciones aritméticas

Los operadores aritméticos realizan operaciones matemáticas, como sumas o restas con operandos. Hay dos tipos de operadores matemáticos: unarios y binarios. Los operadores unarios realizan una acción con un solo operando. Los operadores binarios realizan acciones con dos operandos. En una expresión compleja (dos o más operandos), el orden de evaluación depende de las reglas de precedencia.

(Operadores y expresiones, s. f.)

1.2.1. Operadores unitarios

Los operadores unarios son operadores aritméticos que realizan una acción sobre un solo operando. El lenguaje de script reconoce el operador unario negativo (-).

El operador unario negativo invierte el signo de una expresión, de positivo a negativo o viceversa. El efecto neto es el de multiplicar el número por -1.

(Operadores y expresiones, s. f.)

1.2.2. Operadores binarios

Figura 2. **Tipos de operadores binarios**

Símbolo	Operación	Ejemplo	Descripción
+	Suma	a + b	Sumar los dos operandos
-	Resta	a - b	Restar el segundo operando del primero
*	Multiplicación	a * b	Multiplicar los dos operandos
/	División	a / b	Dividir el primer operando por el segundo
**	Potencia	a ** b	Elevar el primer operando a la potencia del segundo operando
%	Resto	a % b	Dividir el primer operando por el segundo y dar como resultado la parte restante

Fuente: https://www.ibm.com/docs/es/pureapplication-service/2.3.1.0?topic=language-operators-expressions

1.3. Operaciones lógicas

Las operaciones lógicas son expresiones matemáticas cuyo resultado es un valor booleano (verdadero o falso; true o false). Estas expresiones se utilizan principalmente en las estructuras de control.

(Marco, s. f.)

Figura 3. **Tipos de operaciones lógicas**

Ejemplo	Nombre	Resultado
expresión_1 == expresión_2	Igual	**true** si expresión_1 es igual a expresión_2.
expresión_1 === expresión_2	Idéntico	**true** si expresión_1 es igual a expresión_2, y son del mismo tipo. (a partir de PHP 4)
expresión_1 != expresión_2	Diferente	**true** si expresión_1 no es igual a expresión_2.
expresión_1 <> expresión_2		
expresión_1 !== expresión_2	No idénticos	**true** si expresión_1 no es igual a expresión_2, o si no son del mismo tipo. (a partir de PHP 4)
expresión_1 < expresión_2	Menor que	**true** si expresión_1 es estrictamente menor que expresión_2.
expresión_1 > expresión_2	Mayor que	**true** si expresión_1 es estrictamente mayor que expresión_2.
expresión_1 <= expresión_2	Menor o igual que	**true** si expresión_1 es menor o igual que expresión_2.
expresión_1 >= expresión_2	Mayor o igual que	**true** si expresión_1 es mayor o igual que expresión_2.

Fuente: https://www.mclibre.org/consultar/php/lecciones/php-operaciones-logicas.html#:~:text=Las%20operaciones%20lógicas%20son%20expresiones,en%20las%20estructuras%20de%20control.

1.4. Ciclo for

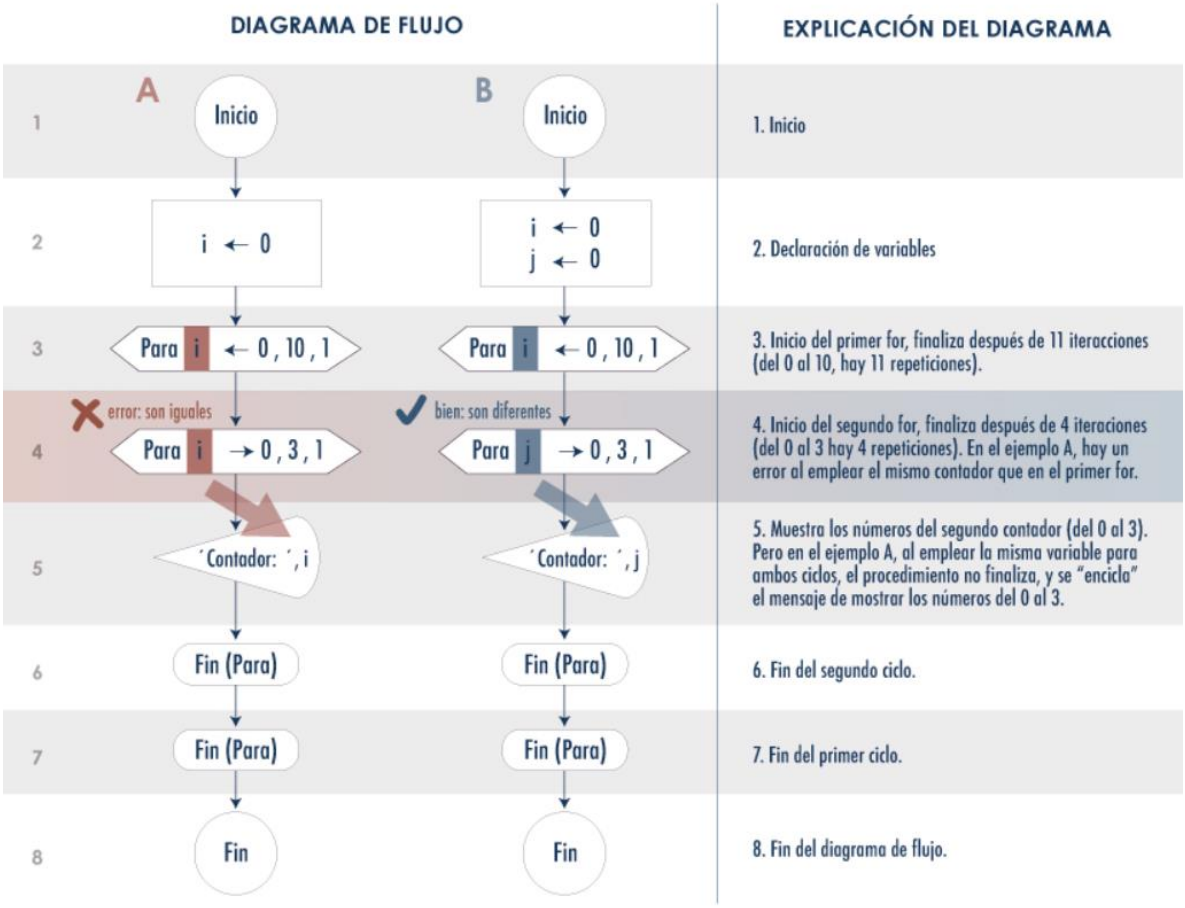
El ciclo for es uno de los más utilizados en programación debido a que permite repetir varias instrucciones n veces. Se emplea en el recorrido de vectores, matrices y estructuras, entre otros.

1.4.1. características

- Siempre se hace uso de una variable (contador) que incrementará su valor automáticamente y ayudará a determinar si se continúa o finaliza el ciclo.
- El contador deberá inicializarse con un valor, generalmente 0 ó 1, dependiendo de lo que se esté realizando.
- Un ciclo puede contener otro ciclo dentro de sí (a esto se le denomina ciclo anidado). Nunca se debe utilizar el mismo nombre de la variable (contador) en ambos ciclos, pues el programa no podrá determinar cuándo se finaliza el ciclo.

(Ciclo For, s. f.)

Figura 4. **Diagrama de flujo de ciclo for**



Fuente: https://multimedia.uned.ac.cr/pem/ciclo_for/pag/caracteristicas.html

1.5. Ciclo while

El ciclo while al cual también se le conoce también como *ciclo controlado por condición inicial*, porque en este tipo de ciclos, se repite un proceso *mientras* una condición sea verdadera.

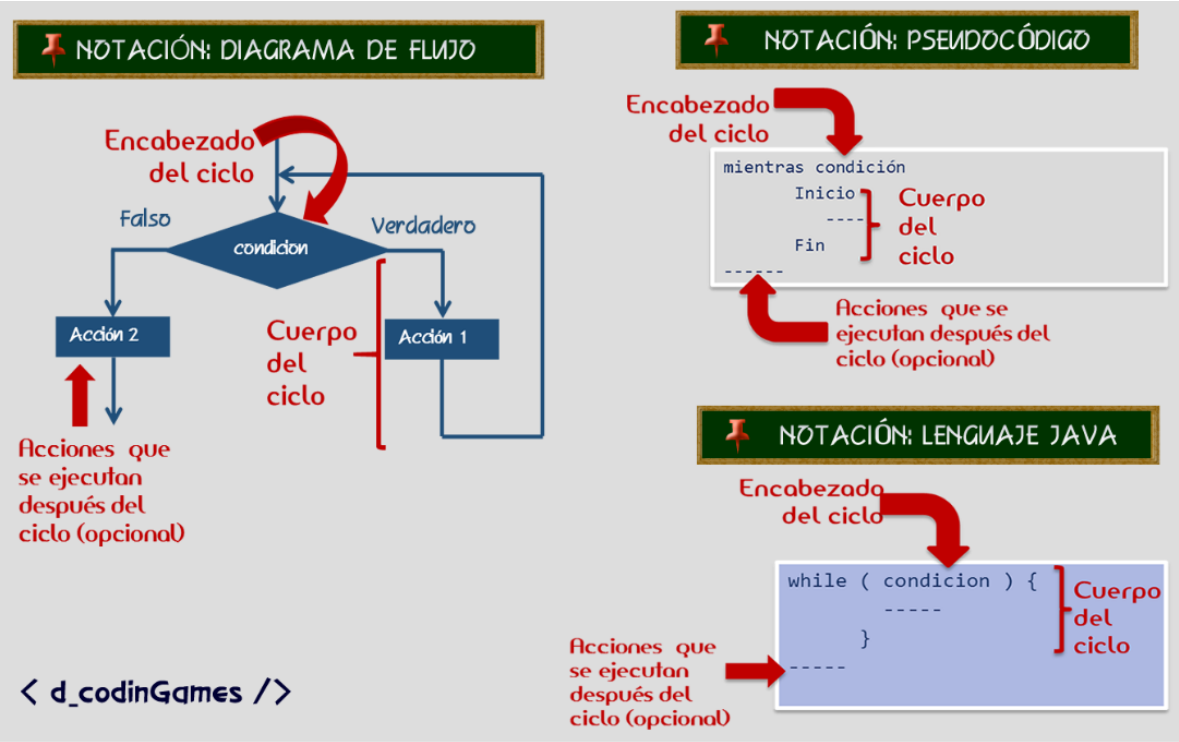
1.5.1. Funcionamiento

- Se inicia evaluando la condición que es parte del encabezado del ciclo.
- Si la condición es verdadera, se dice que el flujo del programa entra al cuerpo del ciclo, y por lo tanto se puede ejecutar la Acción1. En la práctica, el cuerpo del ciclo puede estar conformado por más de una instrucción, pero para simplificar, he escrito solo la palabra Acción1.

- Si se ejecutó la Acción1, se regresa a evaluar nuevamente la condición del ciclo. Si esta sigue siendo verdadera, nuevamente el flujo del programa vuelve a entrar al cuerpo del ciclo.
- Cuando se evalúe la condición y ésta sea falsa, el flujo del programa sale del ciclo y ahora se ejecuta la Acción2, que representa cualquier instrucción (o conjunto de instrucciones) que estén fuera del ciclo y que, por lo tanto, no se repetirán.

(Nieva, 2020)

Figura 5. **Diagrama de flujo de ciclo while**



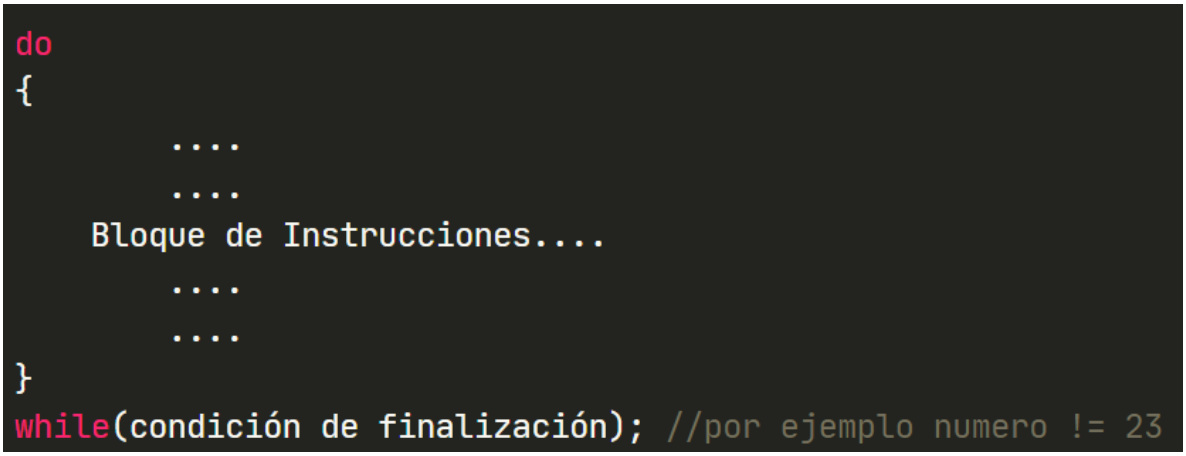
Fuente: https://dcodingames.com/el-ciclo-while/

1.6. Ciclo do while

Los ciclos do-while son una estructura de control cíclica, los cuales nos permiten ejecutar una o varias líneas de código de forma repetitiva sin necesidad de tener un valor inicial e incluso a veces sin siquiera conocer cuando se va a dar el valor final, hasta aquí son similares a los ciclos while, sin embargo el ciclo do-while nos permite añadir cierta ventaja adicional y esta consiste que nos da la posibilidad de ejecutar primero el bloque de instrucciones antes de evaluar la condición

necesaria, de este modo los ciclos do-while, son más efectivos para algunas situaciones específicas. En resumen, un ciclo do-while, es una estructura de control cíclica que permite ejecutar de manera repetitiva un bloque de instrucciones sin evaluar de forma inmediata una condición especifica, sino evaluándola justo después de ejecutar por primera vez el bloque de instrucciones. *(González, 2019)*

Figura 6. **Diagrama de flujo de ciclo while**



Fuente: https://www.programarya.com/Cursos/C++/Ciclos/Ciclo-do-while

1.7. Tipos de casteos

Los casting en programación se utilizan para asegurarse que un dato es de un tipo en concreto. Si es necesario, se convertirá al tipo de dato pedido, pero no sirve en todos los casos, ya que no es un sistema de conversión como tal.

Se pueden realizar casteos de tipo entero a tipo flotante y viceversa.

Se puede realizar casteos de tipo cadena a tipo entero y flotante y viceversa; siempre y cuando la cadena contenga solo números.

(Chichon, 2014)

Conclusiones

Bibliografía

Operadores relacionales. (s. f.). © Copyright IBM Corp. 2014. Recuperado 10 de septiembre de 2022, de https://www.ibm.com/docs/es/iis/11.5?topic=bo-relational-operators

Operadores y expresiones. (s. f.). © Copyright IBM Corp. 2016. Recuperado 10 de septiembre de 2022, de https://www.ibm.com/docs/es/pureapplication-service/2.3.1.0?topic=language-operators-expressions

Marco, B. S. (s. f.). *Operaciones lógicas. PHP. Bartolomé Sintes Marco. www.mclibre.org*. Recuperado 10 de septiembre de 2022, de https://www.mclibre.org/consultar/php/lecciones/php-operaciones-logicas.html#:%7E:text=Las%20operaciones%20l%C3%B3gicas%20son%20expresiones,en%20las%20estructuras%20de%20control.

Ciclo For. (s. f.). Recuperado 10 de septiembre de 2022, de https://multimedia.uned.ac.cr/pem/ciclo_for/pag/caracteristicas.html

Nieva, G. (2020, 23 marzo). *El ciclo while*. dCodinGames. Recuperado 10 de septiembre de 2022, de https://dcodingames.com/el-ciclo-while/

Chichon, P. M. A. (2014, 19 septiembre). *Funciones útiles de cadenas - Programación en C#*. Recuperado 10 de septiembre de 2022, de https://www.aulafacil.com/cursos/programacion/en-c/funciones-utiles-de-cadenas-l17932