

ALGORITMOS

TEma: Condicionales multiples

Ariel Alejandro Calderón | Software | 19-01-2024

# Introducción

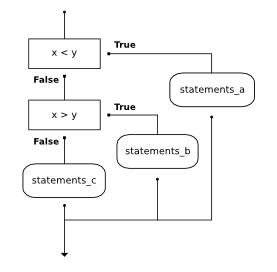
# Las **expresiones condicionales** desempeñan un papel fundamental en el diseño y la ejecución de programas de software, ya que permiten que un programa tome decisiones basadas en la evaluación de condiciones específicas. Su importancia radica en varios aspectos clave:

# **Control de Flujo:** Las expresiones condicionales proporcionan un control de flujo esencial, permitiendo que un programa seleccione diferentes caminos de ejecución en función de condiciones específicas. Esto es fundamental para diseñar algoritmos y lógica de programación que se adapten dinámicamente a diversas situaciones.

# **Adaptabilidad y Flexibilidad:** La capacidad de ejecutar bloques de código condicionalmente permite que los programas se adapten a diferentes escenarios y condiciones del entorno.La flexibilidad resultante facilita la creación de aplicaciones más robustas y versátiles que pueden manejar una variedad de situaciones.

# **Optimización del Rendimiento:** Las expresiones condicionales también se utilizan para optimizar el rendimiento del programa al ejecutar ciertos bloques de código solo cuando sea necesario. Al evitar la ejecución de código innecesario, se mejora la eficiencia y la velocidad del programa.

# **Manejo de Datos de Entrada:** En la validación de datos de entrada, las expresiones condicionales son fundamentales para garantizar que los datos ingresados cumplan con los requisitos especificados. Se utilizan para verificar la validez y consistencia de los datos, lo que contribuye a la seguridad y robustez del programa.

****

# **Toma de Decisiones:**Las expresiones condicionales son esenciales para la toma de decisiones dentro de un programa, lo que permite que la lógica del software se alinee con los requisitos del usuario o del sistema.

# **Manejo de Excepciones:** Las expresiones condicionales permiten que el código responda de manera adecuada ante condiciones inesperadas, mejorando la robustez y la experiencia del usuario.

# **Legibilidad del Código:** Una implementación adecuada de expresiones condicionales contribuye significativamente a la legibilidad del código.

## Condicionales multiples

Las expresiones condicionales múltiples son estructuras que permiten evaluar múltiples condiciones y ejecutar diferentes bloques de código según el resultado de estas evaluaciones. Estos bloques suelen estar compuestos por instrucciones condicionales como **if, else if** y else en muchos lenguajes de programación.

**Ejemplos de uso**

**Ejemplo 1: Evaluación de Rangos**

En este ejemplo, la función **clasificar\_puntuacion** evalúa la puntuación dada y asigna una etiqueta según el rango en el que caiga. Se utilizan múltiples condicionales (if, elif, else) para cubrir distintos intervalos:

### def clasificar\_puntuacion(puntuacion):

### if puntuacion >= 90:

### print("Excelente")

### elif 80 <= puntuacion < 90:

### print("Bueno")

### elif 70 <= puntuacion < 80:

### print("Aceptable")

### else:

### print("Insuficiente")

**Ejemplo 2: Selección de Día de la Semana**

En este caso, la función **obtenerActividadDiaria** sugiere diferentes actividades según el día de la semana. Se utilizan múltiples condiciones para especificar las actividades para días específicos. Como datos de entrada serian unicamente validos cadenas texto que representen los dias de la semana:

### function obtenerActividadDiaria(dia) {

### if (dia === 'Lunes' || dia === 'Miércoles') {

### console.log("Ir al gimnasio");

### } else if (dia === 'Viernes') {

### console.log("Ver una película");

### } else if (dia === 'Domingo') {

### console.log("Descansar");

### } else {

### console.log("Trabajar o hacer actividades normales");

### }

### }

**Ejemplo 3: Validación de Entrada de Usuario**

En este ejemplo en Java, el programa solicita al usuario que ingrese su edad y luego utiliza condicionales múltiples para determinar si es mayor de edad, un adolescente o un niño:

### import java.util.Scanner;

### public class ValidacionEdad {

### public static void main(String[] args) {

### Scanner scanner = new Scanner(System.in);

### 

### System.out.print("Ingrese su edad: ");

### int edad = scanner.nextInt();

### if (edad >= 18) {

### System.out.println("Eres mayor de edad");

### } else if (edad >= 13) {

### System.out.println("Eres un adolescente");

### } else {

### System.out.println("Eres un niño");

### }

### }

### }

Estos son solo algunos ejemplos, y la aplicación de expresiones condicionales múltiples puede variar según el contexto y los requisitos específicos de cada programa. Es importante tener en cuenta la claridad y la eficiencia al utilizar condicionales múltiples para garantizar un código fácil de entender y mantener.

**Bibliografía:**

* [Estructuras-condicionales-multiples - https://tutorialesdeaplicaciones.com](https://tutorialesdeaplicaciones.com/estructuras-condicionales-multiples/)
* [Estructuras condicionales - https://desarrolloweb.com](https://desarrolloweb.com/articulos/2225.php)
* [Ejercicios-condicionales-multiples - https://logicadeprogramacion.neocities.org](https://logicadeprogramacion.neocities.org/sitioweb/ejercicios-condicionales-multiples)