

Programación Orientada a Objetos

TEMA: Paradigmas de la

Programación Orientada a Objetos



INTEGRANTES:

* Ariel Alejandro Calderón
* Ángeles Xiomara Punina

PROBLEMA 1

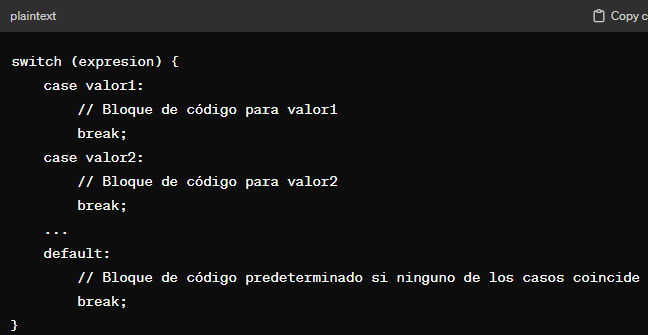
Generar los 10 primeros términos del Fibonacci.

Cada valor posible de una expresión está asociado a un caso específico, y el bloque de código correspondiente a ese caso se ejecuta si el valor de la expresión coincide con el valor especificado en ese caso.

Funcionamiento Básico

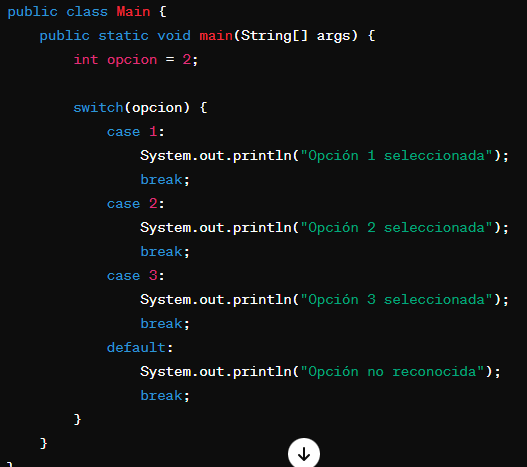
1. Evaluación de expresión: La sentencia *switch* evalúa una expresión y luego ejecuta uno de los varios bloques de código dependiendo del valor resultante de esta expresión.
2. Comparación: Cada *case* en la estructura de *switch* especifica un valor constante o una expresión constante que se compara con el resultado de la expresión evaluada.
3. Coincidencia: Si la expresión *coincide* con un valor en un case, se ejecuta el bloque de código asociado con ese *case*.
4. Break: Después de ejecutar un case, generalmente se usa la palabra clave *break* para salir de la estructura *switch*. Esto evita que se ejecuten los bloques de código de los *case* siguientes.

Sintaxis general

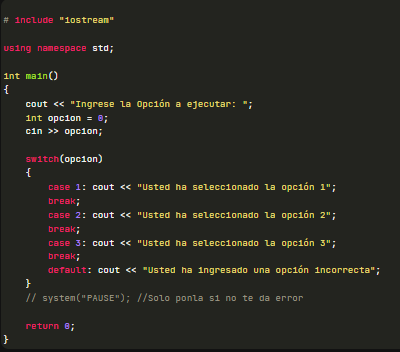


Algunos ejemplos en diferentes lenguajes de programación:

Java



C ++



Aspectos importantes

* Tipos de datos: Dependiendo del lenguaje, los tipos de datos que se pueden usar en los *case* pueden variar. Algunos leguajes solo permiten valores *constantes* (como en C++), mientras que otros permiten *expresiones más complejas* (como en Java).
* Orden de evaluación: Los *case* se evalúan en orden secuencial. Una vez que se encuentra una *coincidencia*, se ejecuta el bloqueo de código correspondiente y luego se sale de la estructura *switch*.
* Uso de break: Olvidar incluir *break* después de una *case* puede llevar a un comportamiento inesperado, ya que la ejecución continuara con los bloqueos de código de los case siguientes hasta encontrar un *break*.
* No se puede repetir valores: No puedes tener dos *case* con el mismo valor en una misma estructura *switch*. Cada valor debe ser único.

Conclusión

La sentencia *switch case* es una herramienta útil para manejar múltiples caminos de ejecución en función de diferentes valores de una expresión en un programa. Es especialmente útil cuando hay una serie de valores posibles y se quiere evitar una cascada de *if-else*.

**Bibliografía**

1. **Libros:**
   * Dheeraj Malhotra, PhD & Neha Malhotra, PhD. (2020). *Data Structures and Program Design Using JAVA* (pp. 24-28). Mercury Learning and Information.  
     Descripción: Se consultaron las secciones que explican los conceptos básicos de la Programación Orientada a Objetos, incluyendo objetos, clases y encapsulación.
2. **Recursos Web:**
   * GeeksforGeeks. (n.d.). Object-oriented programming (OOP) in Java. GeeksforGeeks. Retrieved September 5, 2024, from https://www.geeksforgeeks.org/object-oriented-programming-oops-concept-in-java/  
     Descripción: Artículo que explica los conceptos de encapsulación, herencia, polimorfismo y abstracción con ejemplos en Java.