

Conceptos y Paradigmas de Lenguajes de Programación - Práctica 9

EJERCICIO 1

Excepción

Es una condición inesperada o inusual que surge durante la ejecución del programa y no puede ser manejada en el contexto local. Condición anómala e inesperada que necesita ser controlada.

EJERCICIO 2

Para que un lenguaje trate excepciones debe proveer mínimamente:

- Un modo de definirlas
- Una forma de reconocerlas
- Una forma de lanzarlas y capturarlas
- Una forma de manejarlas especificando el código y respuestas
- Un criterio de continuación

No, no todos los lenguajes lo proveen, por ejemplo **C** estándar **no provee** manejo de excepciones.

EJERCICIO 3

Cuando un lenguaje de programación **no proporciona un mecanismo de manejo de excepciones**, puede ser más difícil y menos estructurado manejar errores y situaciones excepcionales en el código. Sin un manejo adecuado de excepciones, los errores pueden propagarse sin control, lo que puede llevar a un comportamiento inesperado y a una mayor dificultad para depurar el código.

Aunque no se disponga de un mecanismo de manejo de excepciones incorporado en el lenguaje, es posible simularlo utilizando técnicas alternativas. Una forma común de simular el manejo de excepciones es mediante el **uso de estructuras de control condicionales y funciones de retorno de errores**.

Es importante tener en cuenta que este enfoque simulado del manejo de excepciones puede resultar más propenso a errores y puede llevar a un código **más complicado y difícil de mantener** en comparación con un lenguaje que ofrece un manejo de excepciones nativo. Además, la simulación no proporcionará todas las funcionalidades avanzadas de los mecanismos de manejo de excepciones, como la capacidad de capturar excepciones específicas o realizar acciones de limpieza antes de propagar el error.

EJERCICIO 4 - 14