

Practica 2 Tipos de datos y Disciplina de tipos

Eduardo Alcaraz

February 12, 2024

Contents

La disciplina de tipos se refiere al conjunto de reglas que definen cómo los tipos de datos se utilizan y se gestionan en un lenguaje de programación. Esta disciplina es crucial para entender cómo los lenguajes de programación clasifican y operan con los datos, y cómo esto afecta la seguridad, la corrección y la eficiencia del código. Hay varios aspectos clave que definen la disciplina de tipos en los lenguajes de programación:

- **Tipado Fuerte vs. Tipado Débil**

- **Tipado Fuerte:** En un lenguaje de tipado fuerte, las conversiones de tipo son más controladas y restringidas. Esto significa que el lenguaje no permitirá operaciones que intenten mezclar tipos incompatibles sin una conversión explícita. Este enfoque puede ayudar a prevenir errores en tiempo de ejecución, ya que todos los tipos son verificados de manera estricta.
- **Tipado Débil:** Los lenguajes de tipado débil permiten más flexibilidad en las conversiones de tipos, a menudo realizando conversiones implícitas entre tipos incompatibles. Esto puede hacer que el código sea más fácil de escribir, pero también puede introducir errores sutiles y difíciles de detectar.

- **Tipado Estático vs. Tipado Dinámico**

- **Tipado Estático:** En un lenguaje de tipado estático, el tipo de cada variable y expresión debe ser conocido en tiempo de compilación. Esto permite que el compilador detecte errores de tipo antes de la ejecución del programa, lo cual puede mejorar la seguridad y eficiencia del código.

- **Tipado Dinámico:** Los lenguajes de tipado dinámico determinan los tipos de las variables en tiempo de ejecución. Esto ofrece una mayor flexibilidad y puede hacer que el código sea más fácil de escribir y leer, pero a costa de una posible reducción en el rendimiento y aumentando el riesgo de errores de tipo en tiempo de ejecución.

- **Inferencia de Tipos**

Algunos lenguajes de programación pueden inferir el tipo de una variable automáticamente basándose en el contexto en el que se usa. Esto puede reducir la necesidad de declaraciones de tipo explícitas, simplificando el código sin sacrificar la seguridad del tipado estático.

- **Sistemas de Tipos**

El sistema de tipos de un lenguaje de programación es el conjunto de reglas que define cómo se pueden combinar y operar los tipos. Esto incluye:

- **Compatibilidad de Tipos:** Cómo los tipos se relacionan entre sí y bajo qué condiciones se consideran compatibles.
 - **Polimorfismo:** La capacidad de abstraer operaciones en diferentes tipos de manera que puedan trabajar con datos de varios tipos.
 - **Tipos Genéricos (o Paramétricos):** Permiten la definición de estructuras de datos o funciones que pueden operar sobre cualquier tipo, definido en el momento de la utilización.

Dado los siguientes lenguajes:

- Python
- C
- Java
- Perl
- Go
- Rust

Se requiere resolver los siguientes programas,

- Calcular el factorial de un numero

- Calcular el fibonacci de un numero
- Validar si una palabra es capicúa o no

Para cada código se debe de identificar los siguientes puntos.

- Tipo de dato que maneja el lenguaje
- La disciplina de tipos
- El paradigma del lenguaje utilizado
- La estructura que lleva las funciones o métodos