**Nombre:** Jaime Barrera  
**Gmail:** [Jaenba10@gmail.com](mailto:Jaenba10@gmail.com) **Grupo:** T2

**Compiladores e Intérpretes:**

Los compiladores y los intérpretes son dos tipos de programas utilizados para ejecutar código. Los compiladores traducen todo el código fuente al lenguaje de máquina de una vez, generando un archivo ejecutable, mientras que los intérpretes traducen y ejecutan el código línea por línea.

**Tipo fuerte y tipo débil:**

Los lenguajes de tipado fuerte requieren que los tipos de datos sean estrictamente adheridos, lo que significa que las conversiones entre tipos deben ser explícitas. En contraste, los lenguajes de tipado débil permite conversiones implícitas entre tipos de datos, lo que puede conducir a comportamientos inesperados o errores si no se manejan adecuadamente.

Las principales diferencias entre los lenguajes de tipado fuerte y débil son:

1. **Coerción de tipos:** En los lenguajes de tipado fuerte, la coerción de tipos (la conversión automática entre tipos) es mínima o inexistente, lo que significa que se requiere una conversión explícita entre tipos de datos diferentes. En los lenguajes de tipado débil, la coerción de tipos es más común y puede ocurrir de manera implícita, lo que puede conducir a comportamientos inesperados.

2. **Seguridad de tipos:** Los lenguajes de tipado fuerte tienden a ser más seguros en cuanto a la prevención de errores relacionados con el tipo de datos, ya que las conversiones de tipo deben ser explícitas y se aplican reglas estrictas sobre la compatibilidad de tipos. En los lenguajes de tipado débil, la flexibilidad en la coerción de tipos puede llevar a errores sutiles que pueden ser difíciles de detectar.

3. **Claridad y mantenibilidad del código:** Debido a la necesidad de conversiones explícitas, el código en lenguajes de tipado fuerte tiende a ser más claro y comprensible, ya que las operaciones con tipos de datos son explícitas y fácilmente identificables. En contraste, en los lenguajes de tipado débil, donde las conversiones de tipos pueden ocurrir de manera implícita, el código puede ser menos claro y más propenso a errores de interpretación.

En resumen, los lenguajes de tipado fuerte priorizan la seguridad y la claridad del código al requerir conversiones explícitas y ser estrictos en la compatibilidad de tipos, mientras que los lenguajes de tipado débil ofrecen más flexibilidad pero pueden aumentar el riesgo de errores y dificultar la comprensión del código.