Lenguajes compilados e interpretados.

Un compilador recoge un fichero y lo convierte en un lenguaje (puede ser código máquina o código intermedio.

Un lenguaje compilado se compila a lenguaje de máquina.

Un interpretado no.

Codigo fuente – Aquí trabaja el compilador – Codigo final

1 Analizador léxico

El analizador léxico de un compilador va leyendo letra por letra y lo tokeniza. Lo divide carácter por carácter.

Ejemplo

VAR VARIABLE = 10

V – LETRA V

A—LETRA A

…

= ---- SIMBOLO =

1--- NUMERO

0 ---- NUMERO

VAR –(Palabra reservada VAR)

Variable – (identificador)

= --- Simbolo de asignación

10 Numero entero

2 Fase de análisis sintáctico

Analiza si tiene coherencia lo escrito.

3 Genera Código Intermedio

Ejemplo MSI de .net.

Es un código que simplifica el lenguaje humano. Pero no llega a un lenguaje se máquina aún.

Abstract Syntax tree (Arbol sintáctico abstracto):

Es una estructura en forma de árbol de nuestro código a nivel de estructura de datos.

Ejemplo:

2 + 4 sería:

+

2 4

4 Optimización del código intermedio

Se puede generar un código mas óptimo del que hemos escrito.

Por ejemplo eliminar parte del código que nunca se utilizaría.

5 Fase de Generación de Código.

Genera Código máquina o para ser interpretado.

Un lenguaje Compilado se ejecuta directamente por el SO. Por ejemplo C, C++, RUST, GO, Por excelencia el Ensamblador.

Los lenguajes interpretados se ejecuta a través de una m+aquina virtual o un intérprete.

Por ejemplo JAVA.

Se ejecuta con la JVM corriendo por debajo.

En Java se genera un código llamado BYTECODE por el compilador que es el que interpreta la JVM.

Interpretados: PHP, Python y JAVA.

Hay lenguajes compilados y a la vez interpretados:

C Sharp

Se genera un código intermedio que se llama Msil. Y se interpreta. Pero el intérprete por su cuenta decide compilar las partes que del código que se ejecutan muchas veces. Esta técnica se llama Just in Time: Compilador Justo a tiempo.

C# -- MSIL (Microsoft Intermediate Language)

--- Compila partes que se repiten (JIT Compiler).

Los lenguajes compilados son mucho mas rápidos. Pero hay que generar código para cada tipo de plataforma.

Los compiladores no sirven solo para lenguajes. Algunos emuladores compilan al código de nuestra máquina.

El interpretado es mas sencillo de depurar.

Otra ventaja es que se puede ejecutar en cualquier maquina. Ya que paso el código intermedio y el intérprete se encarga.