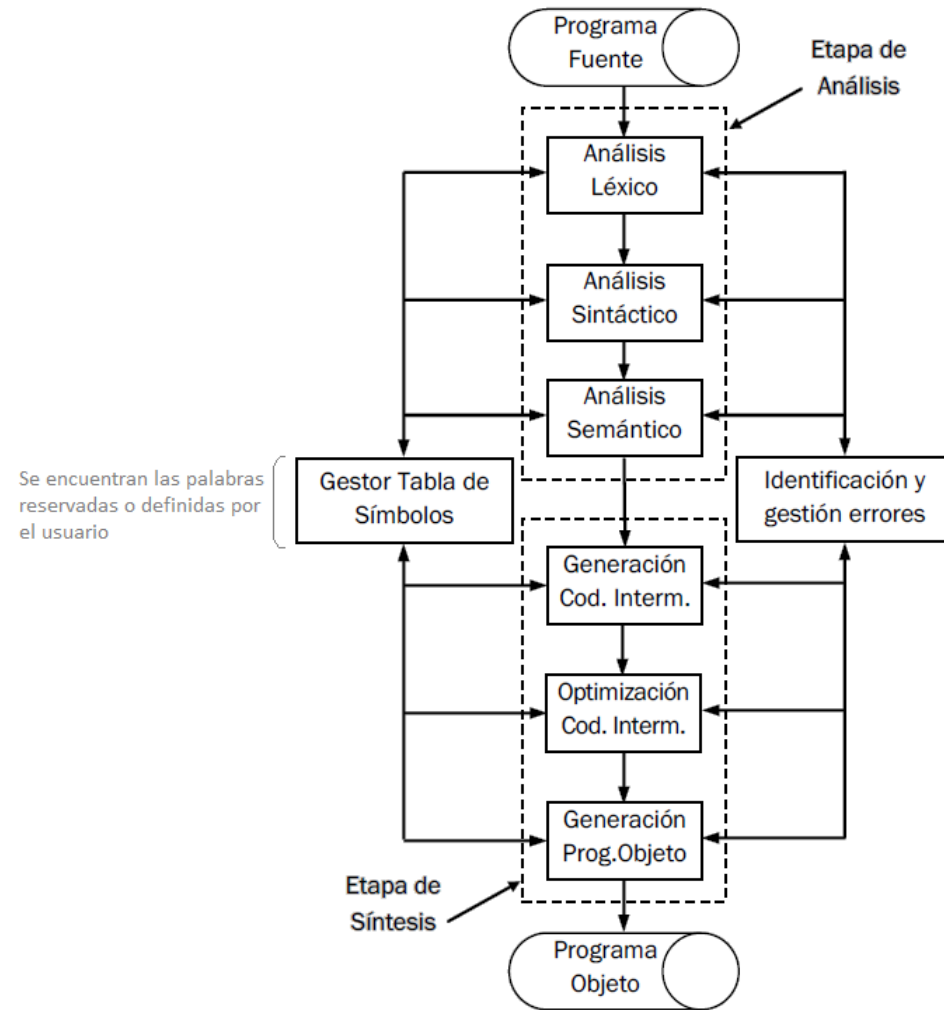


Estructura de compiladores e intérpretes



Programa fuente: Es aquel que se crea con un lenguaje de alto nivel

Programa objeto: Es el lenguaje que interpreta la maquina

Etapa de análisis: Se divide el programa fuente en sus elementos componentes y se crea una representación intermedia del mismo
Se busca determinar que el programa esté bien escrito

Análisis léxico: Se analizan las palabras con las cuales está formado el programa fuente. Pueden ser palabras reservadas o palabras definidas por el usuario (variables, nombres, subrutinas, etc.).

Se analizan las palabras que están en el Gestor de Tablas o Símbolos. Si la palabra se encuentra en esta tabla se va a tomar como un componente léxico válido o como un lexóma. Si no se encuentra la palabra se va a identificar algún tipo de error léxico para indicar que esa palabra no pertenece al lenguaje.

Análisis sintáctico: Se analizan las frases. Se determina si estas están sintácticamente correctas. Por ejemplo, `read(x)` es válido, pero `x read()` no lo es. Para esto se utiliza un autómata pila llamado analizador sintáctico. La estructura de memoria de este analizador tiene formato de pila.

Análisis semántico: Se realiza una identificación del tipo de variables que pueden ser fruto de los procesos que se hayan dado en el análisis sintáctico y determina si las frases se pueden ejecutar. Por ejemplo, la suma de un valor numérico con un valor alfanumérico no se puede ejecutar y constituye un error semántico. Se encarga de determinar si la frase que está sintácticamente correcta se ejecute como se espera.

Etapa de síntesis: Se reconstruye el programa objeto a partir de esa representación intermedia

Generación de código intermedio: Adapta el código en el que está escrito el programa fuente al código en el cual va a trabajar el procesador

Optimización de código: Optimiza el código intermedio mediante recursos como librerías

Generación de programa objeto: Genera el programa que ejecutará el procesador