Introducción al Diseño de Compiladores

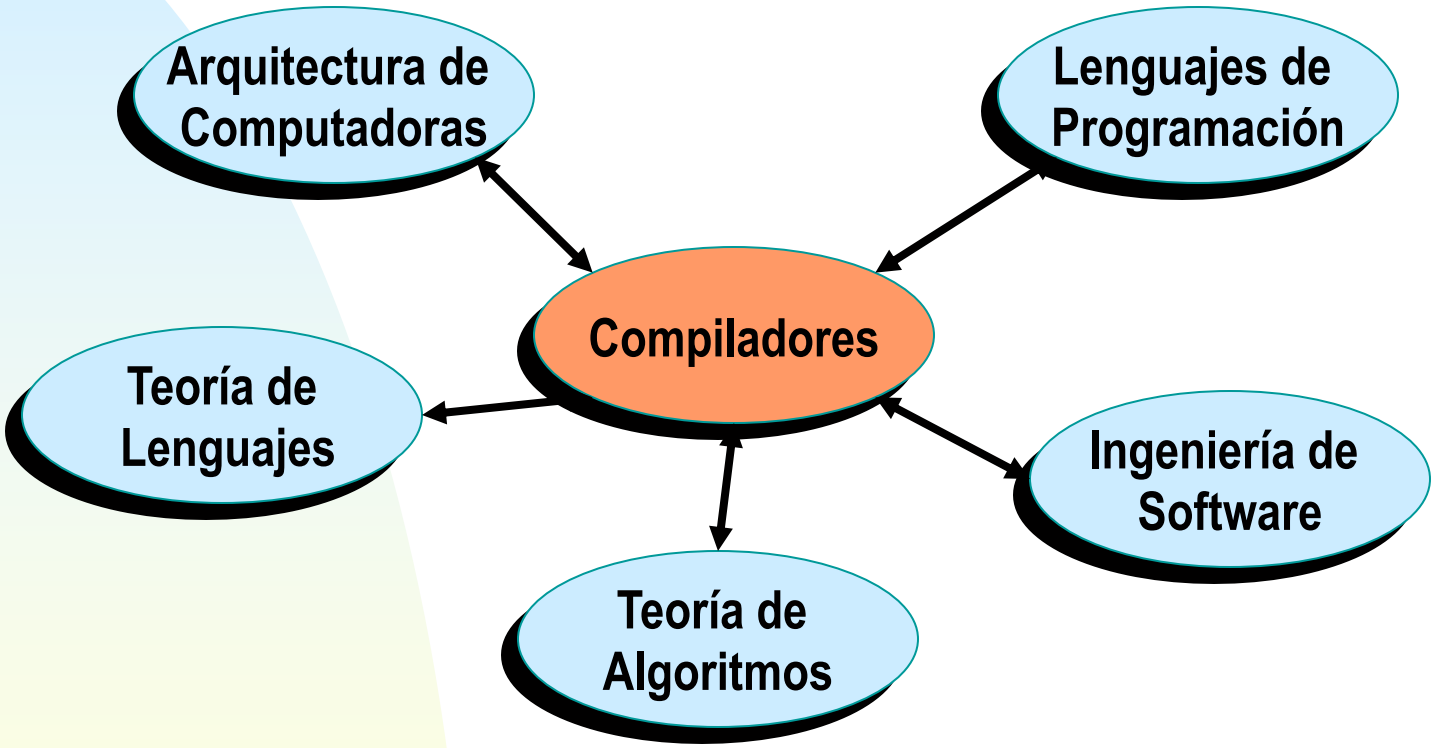
ASIGNATURA: Programación de Sistemas de Base I

CONTENIDOS

- Tema 1.- Introducción a la Compilación
- Tema 2.- Lenguajes, autómatas y gramáticas
- Tema 3.- Análisis léxico
- Tema 4: Tablas de Símbolos
- Tema 5.- Análisis sintáctico
- Tema 6.- Análisis semántico

INTRODUCCIÓN

Conceptos relacionados



Con algunas técnicas básicas de escritura de compiladores se pueden construir traductores para una gran variedad de lenguajes y máquinas

Compiladores

Un compilador es un programa que lee un programa escrito en algún lenguaje (fuente) y lo traduce a un programa equivalente en otro lenguaje (objeto), y además informa al usuario sobre la presencia de errores en el programa de entrada.

CLASIFICACIÓN GENERAL

- De una pasada o de múltiples pasadas
- De carga y de ejecución
- De depuración o de optimización

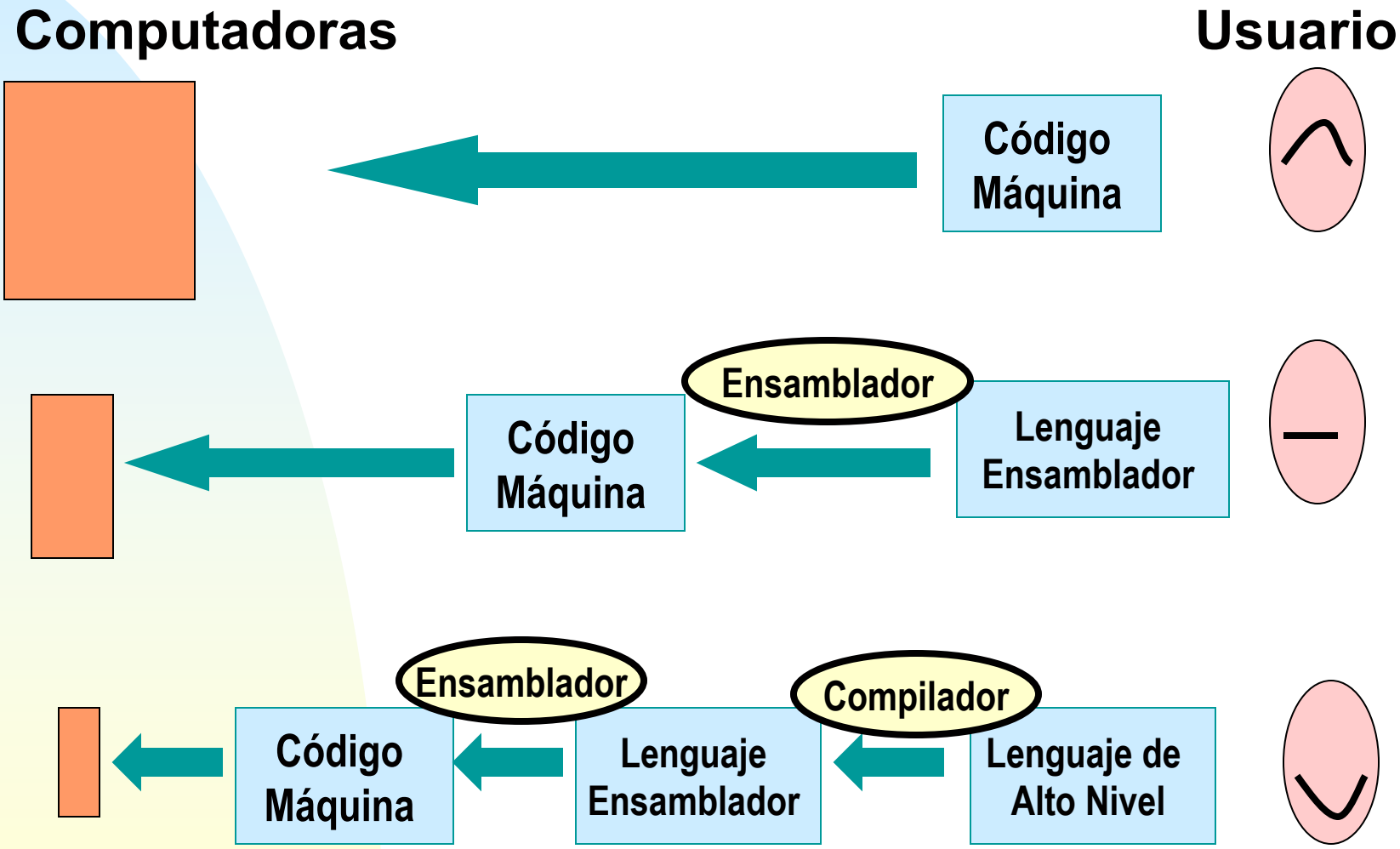
CLASIFICACIÓN GENERAL

- De una pasada o de múltiples pasadas
- De carga y de ejecución
- De depuración o de optimización

HISTORIA

- Experimentación relacionada a traducción de fórmulas
- 1950: difícil escritura
- Primer FORTRAN: 1954
- Hoy: técnicas sistemáticas, lenguajes de implementación, entornos de programación y herramientas de software

HISTORIA


Computadoras
Usuario
Código Máquina
Ensamblador
Código Máquina
Lenguaje Ensamblador
Ensamblador
Compilador
Código Máquina
Lenguaje Ensamblador
Lenguaje de Alto Nivel

HOY…. Y A FUTURO

- El Diseño de un compilador surge como resultado de:
  - ✓ Desarrollo de un nuevo lenguaje de programación
  - ✓ Adición de extensiones a los ya existentes
  - ✓ Explotación de las características del hardware
- A futuro:
  - ✓ Extensión para el cómputo paralelo y distribuido

TIPOS DE SISTEMAS DE COMPILACIÓN

- **ENSAMBLADOR**

  Traducen programas escritos en lenguaje ensamblador a código máquina

- **COMPILADOR**

  Traducen programas escritos en lenguaje de alto nivel a código intermedio o a código máquina

TIPOS DE SISTEMAS DE COMPILACIÓN

- **ENSAMBLADOR**

  Traducen programas escritos en lenguaje ensamblador a código máquina

- **COMPILADOR**

  Traducen programas escritos en lenguaje de alto nivel a código intermedio o a código máquina

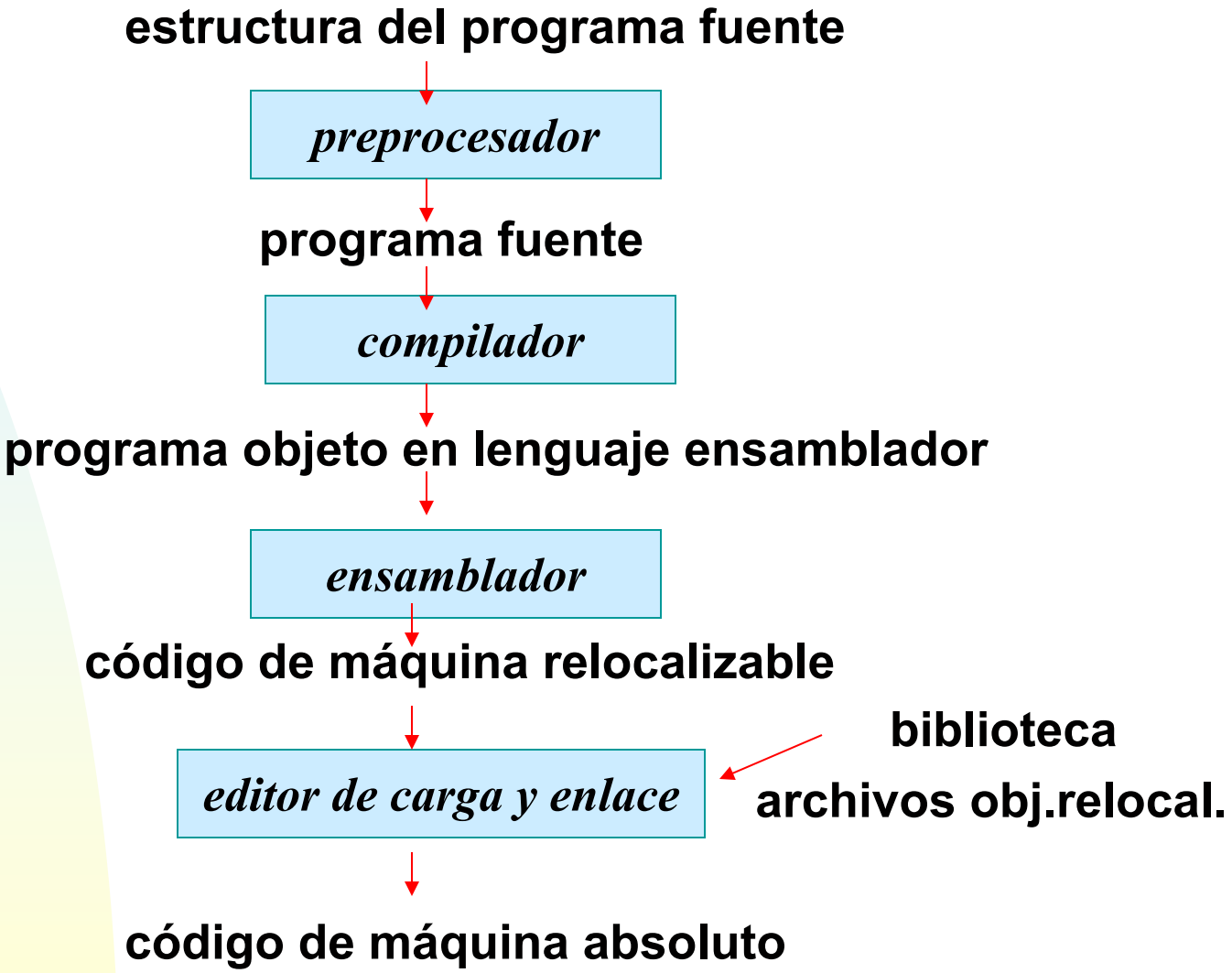
- **INTERPRETE**

  No genera código objeto, analiza y ejecuta directamente cada proposición del Programa Fuente (PF)

- **PREPROCESADOR**

  Sustituyen macros, incluyen archivos o extensión del lenguaje.

SISTEMA PARA PROCESAMIENTO DE UN LENGUAJE



PARTES DE LA COMPILACIÓN

- **ANÁLISIS (Etapa Inicial):**

  Divide al PF en sus elementos componentes y crea una representación intermedia.

  Se determinan las operaciones y se registran en una estructura de árbol (ej. árbol sintáctico)

PARTES DE LA COMPILACIÓN

- **ANÁLISIS (Etapa Inicial):**

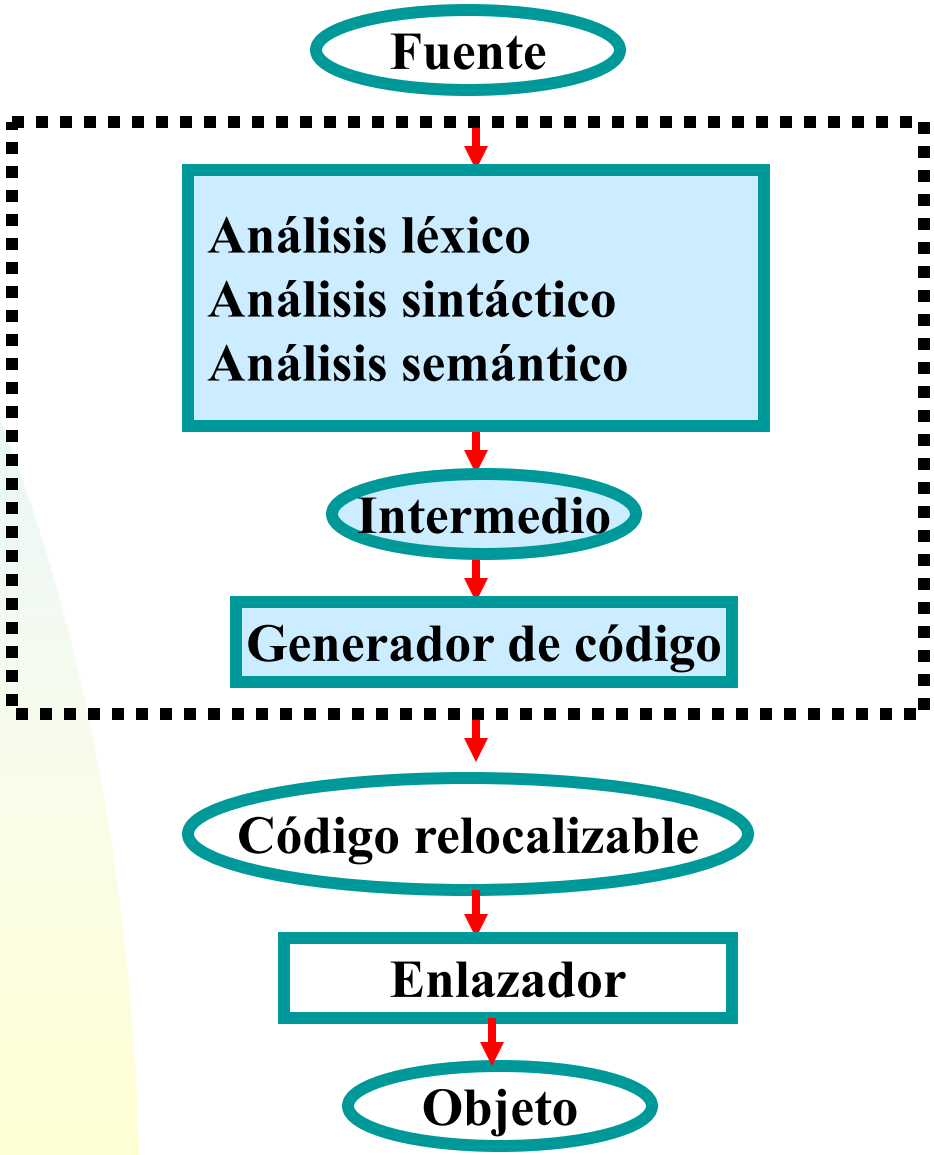
  Divide al PF en sus elementos componentes y crea una representación intermedia.

  Se determinan las operaciones y se registran en una estructura de árbol (ej. árbol sintáctico)

- **SÍNTESIS (Etapa Final):**

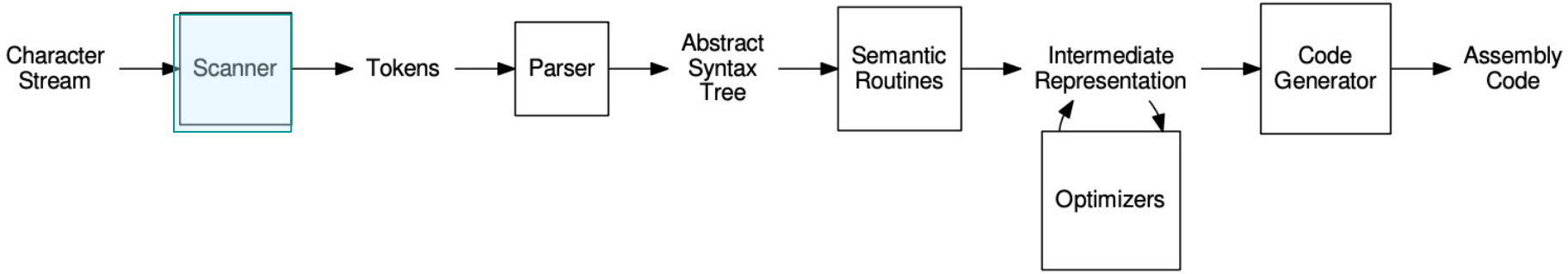
  Construye el PO deseado a partir de la representación Intermedia (requiere técnicas más especializadas)

UN AMBIENTE GENERAL DE COMPILACIÓN



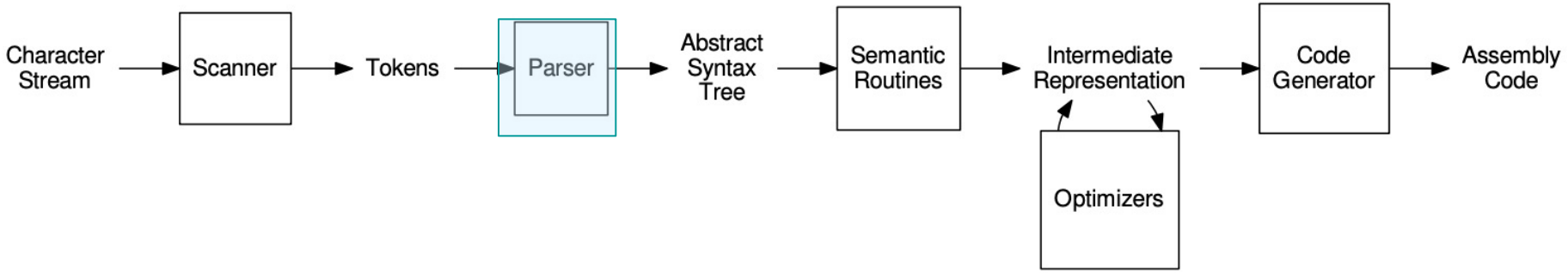
Más:
Sistemas de edición y depuración

Etapas compilador Unix



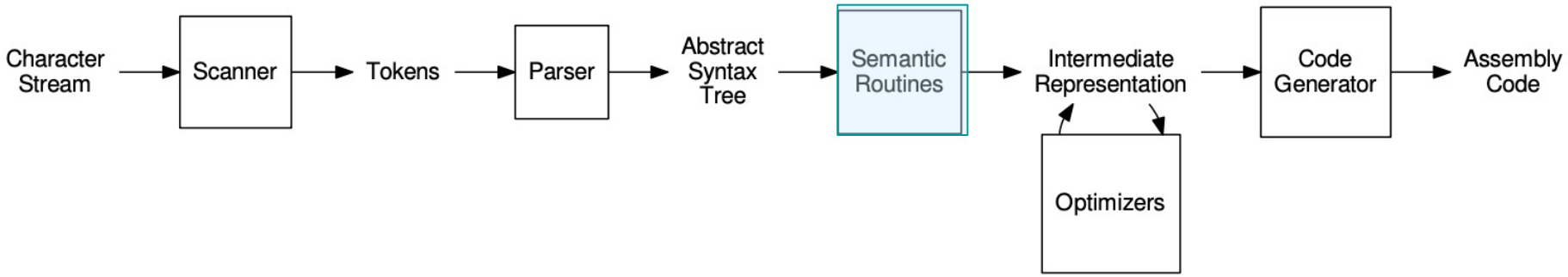
- El Scanner consume el texto simple de un programa, y agrupa los caracteres individuales para formar *tokens* completos.

Etapas compilador Unix



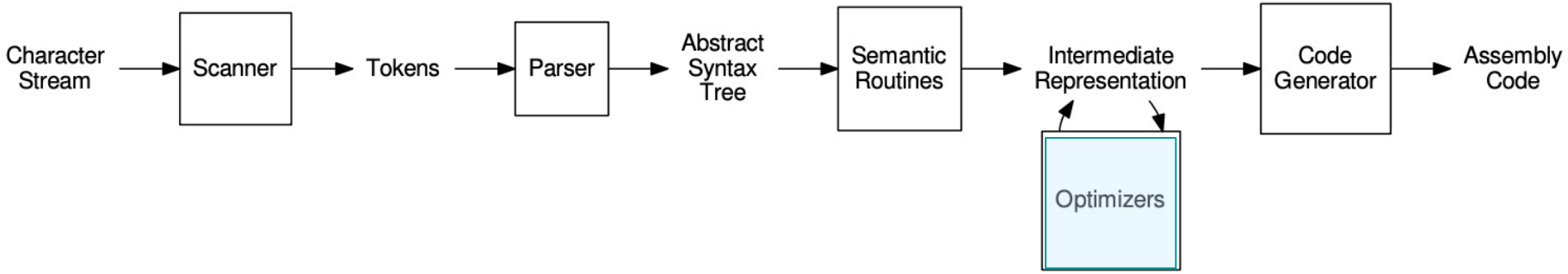
- El parser consume tokens y las agrupa en declaraciones y expresiones completas, de manera similar a como se agrupan las palabras en frases en un lenguaje natural. (Guiado por gramática)

Etapas compilador Unix



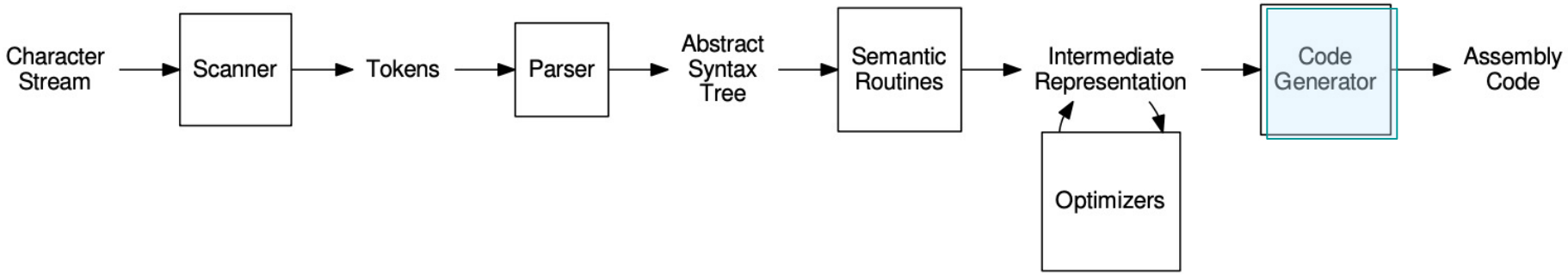
- Deriva un significado adicional (semántica) sobre el programa a partir de las reglas del lenguaje y la relación entre los elementos del programa

Etapas compilador Unix



- Se pueden aplicar uno o más optimizadores a la representación intermedia, para que el programa sea más pequeño, más rápido o más eficiente.

Etapas compilador Unix



- Finalmente, un generador de código consume el IR optimizado y transforma en un programa concreto de lenguaje ensamblador.

Actividad 1

- Compilar programa en C
- Utilizar GCC
  - Disponible por defecto en Linux
  - Instalable en Windows con un port
- Revisar paso por paso:
  - Preprocesamiento
  - Compilador
  - Enlazador
- Busque la gramática formal usada por tres lenguajes