



TECNOLÓGICO  
NACIONAL DE MÉXICO®



GOBIERNO DE  
**MÉXICO**



*Fecha:* 3 de marzo de 2024

# **INSTITUTO TECNOLÓGICO NACIONAL DE MÉXICO**

## **Materia:**

Lenguajes Y Automatas I

## **Trabajo:**

*Fases de un compilador*

## **Alumno:**

José Pablo Martínez González

## **Grupo:**

A

## **Numero de control**

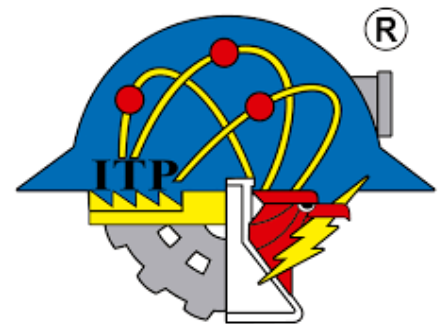
22200194

## **Carrera**

Ingeniería en Sistemas Computacionales

## **Docente**

Baume Lazcano Rodolfo



# Fases de un compilador

```
graph TD; Root[Fases de un compilador] --> Lexical[Análisis léxico]; Root --> Syntactic[Análisis sintáctico]; Root --> Semantic[Análisis semántico]; Root --> Optimization[Optimización de código]; Root --> Intermediate[Generación de código intermedio]; Root --> Final[Generación de código final];
```

## Análisis léxico

Lee el archivo fuente carácter por carácter y forma grupos de caracteres con un significado léxico mínimo, se los denominan tokens y tienen una entidad única.

## Análisis sintáctico

En esta fase, los componentes léxicos se agrupan en fases gramaticales que el compilador utiliza para sintetizar la salida.

## Análisis semántico

Esta fase se hace al mismo tiempo que el sintáctico, ya que introduce rutinas semánticas que intentaban encontrar errores de significado, reuniendo información sobre los tipos de datos del programa fuente.

## Generación de código intermedio

Es un código intermedio del código fuente para el equipo de destino es generado por el compilador después del análisis semántico hace que sea más fácil de traducir el lenguaje en la máquina destino.

## Optimización de código

Esta fase se encarga de optimizar, eliminando código innecesario y organizando la secuencia de declaraciones con el fin de acelerar la ejecución del programa sin desperdiciar de recursos.

## Generación de código final

En esta fase, el generador de código optimizado la representación del código intermedio y la asigna a la máquina de destino, traduciendo el código intermedio en una secuencia.