

# COMPILADOR Y FASES

```
graph TD; A[COMPILADOR Y FASES] --> B[1. ANÁLISIS LÉXICO]; A --> C[2. ANÁLISIS SINTÁCTICO]; A --> D[3. ANÁLISIS SEMÁNTICO]; A --> E[4. GENERACIÓN DE CÓDIGO INTERMEDIO]; A --> F[5. OPTIMIZACIÓN DE CÓDIGO];
```

## 1. ANÁLISIS LÉXICO

Se encarga de dividir el código fuente en unidades léxicas llamadas tokens, como palabras clave, identificadores, operadores, etc.

Un compilador es un programa informático que traduce un código fuente escrito en un lenguaje de programación de alto nivel a un código objeto o código ejecutable que puede ser comprendido y ejecutado por un ordenador.

## 2. ANÁLISIS SINTÁCTICO

Examina la estructura gramatical del código y construye un árbol sintáctico que representa la jerarquía de las expresiones y declaraciones.

## 3. ANÁLISIS SEMÁNTICO

Verifica el significado del código en términos de su lógica y asegura que cumpla con las reglas semánticas del lenguaje.

## 4. GENERACIÓN DE CÓDIGO INTERMEDIO

Produce una representación intermedia del código fuente que facilita el análisis y la optimización.

## 5. OPTIMIZACIÓN DE CÓDIGO

Mejora el rendimiento del código mediante transformaciones que no cambian su comportamiento.

# COMPILADOR Y FASES

```
graph TD; A[COMPILADOR Y FASES] --- B[6. GENERACIÓN DE CÓDIGO]; A --- C[7. OPTIMIZACIÓN DE CÓDIGO FINAL];
```

## 6. GENERACIÓN DE CÓDIGO

Convierte el código intermedio en código máquina específico para la plataforma de destino.

**DIEGO QUINTOS CABEZA**  
**21200625**

## 7. OPTIMIZACIÓN DE CÓDIGO FINAL

Realiza optimizaciones específicas para la plataforma de destino antes de generar el código ejecutable final.

## Conclusión

Los compiladores son herramientas esenciales en el ciclo de vida del desarrollo de software, permitiendo a los programadores centrarse en la lógica y la estructura del código en lugar de preocuparse por los detalles de la ejecución a nivel de máquina.