



# Programación Orientada a Objetos en un Compilador

Exploraremos cómo los compiladores traducen conceptos de POO a instrucciones de bajo nivel en RISC-V. Esta comprensión es crucial para el desarrollo de software eficiente.

 por Henry Mendoza

# Conceptos Fundamentales de POO

## 1 Propiedades

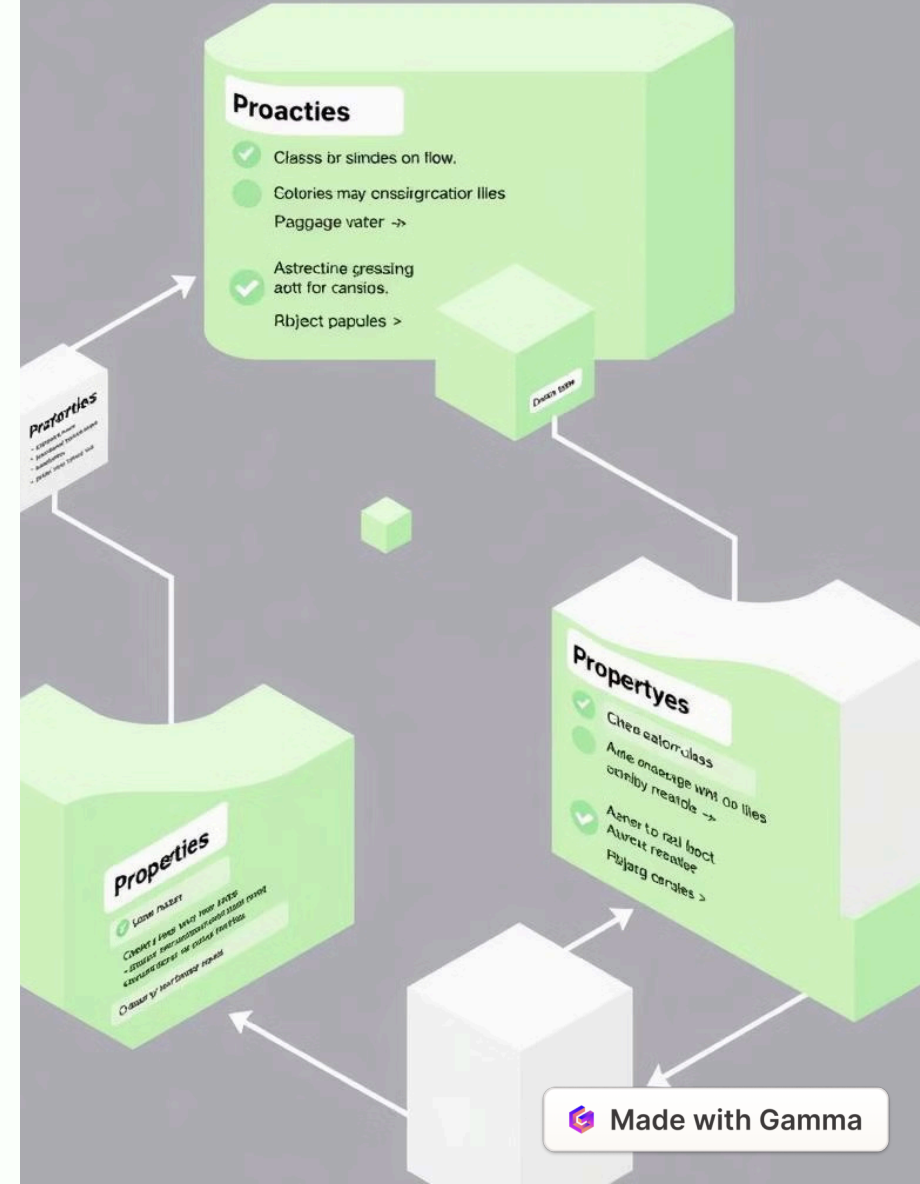
Atributos que definen el estado de un objeto.

## 2 Métodos

Funciones que definen el comportamiento de un objeto.

## 3 Estructura

Organización de propiedades y métodos en una clase.



# Representación de Objetos en Memoria

1

## Asignación

El compilador reserva espacio en memoria para las propiedades del objeto.

2

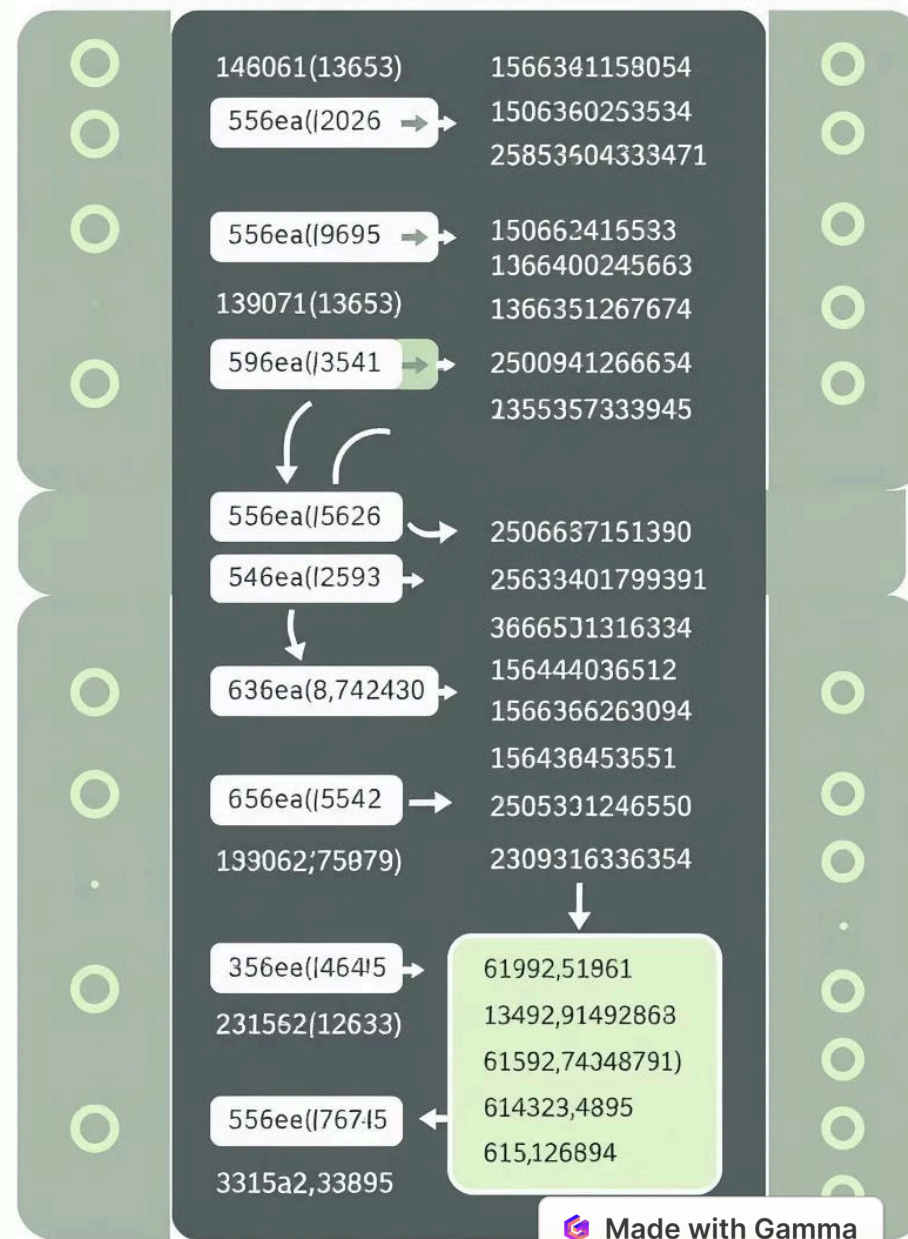
## Organización

Las propiedades se almacenan en estructuras de datos específicas.

3

## Acceso

Se utilizan punteros para acceder a las propiedades en ensamblador.



```
beces ttlt ,a, bolnd riul (pl)  
ftert (right arolecv(to rudmst toce)
```

```
feecting insuaribby: pril  
} tilest arcemos by; function call;
```

# Implementación de Métodos en Ensamblador

1

## Traducción

El compilador convierte los métodos en funciones de ensamblador.

2

## Argumentos

Los parámetros se pasan utilizando registros y la pila.

3

## Invocación

Se utiliza un esquema consistente para llamar y retornar de métodos.

# Ejemplo Práctico de Invocación de Métodos

## Código de Alto Nivel

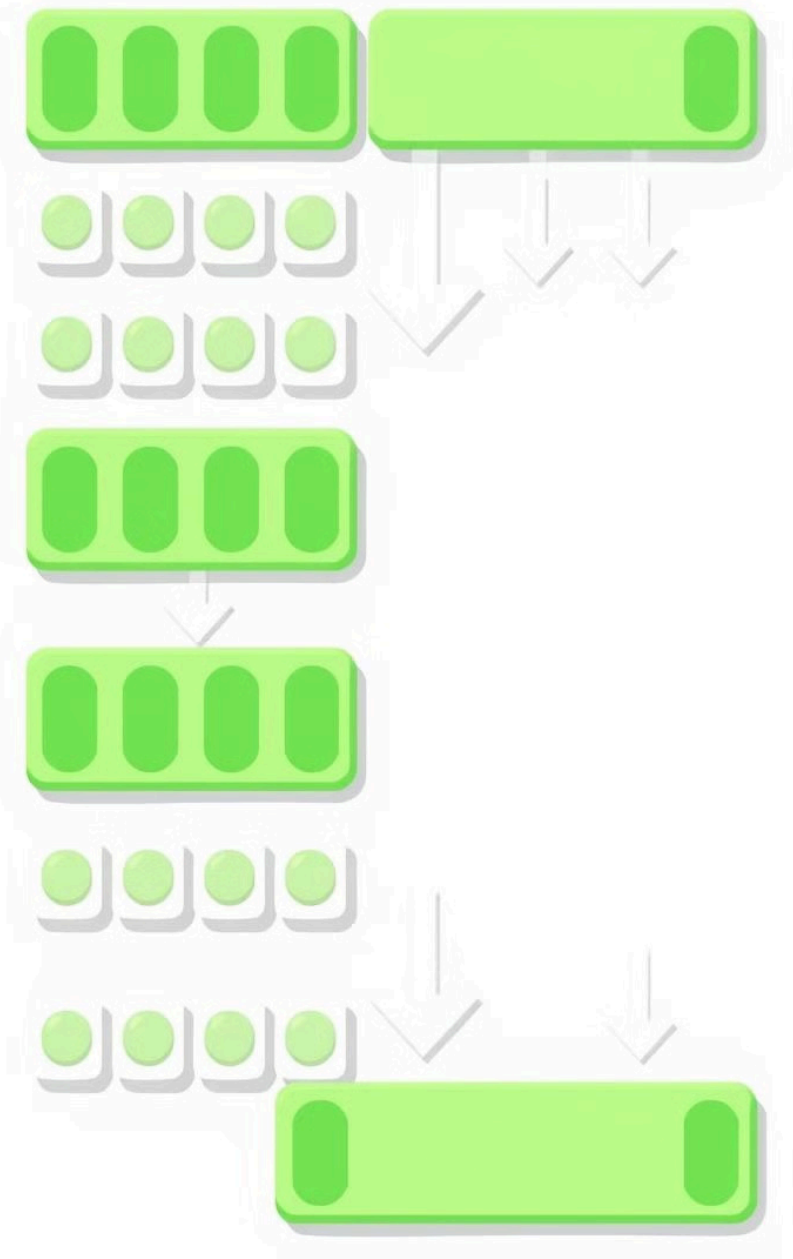
Definición de una clase simple con un método.

## Traducción a RISC-V

Código ensamblador resultante para la invocación del método.

## Paso de Objeto

Cómo se pasa la referencia al objeto en los registros.



# Manejo del Heap en P00

## Asignación Dinámica

Instrucciones para reservar espacio en el heap para objetos.

## Liberación

Gestión de la memoria para evitar fugas y optimizar el rendimiento.

## Eficiencia

Estrategias para minimizar la fragmentación y el uso de memoria.



# Optimización en la Gestión de Objetos y Memoria



## Uso Eficiente de Registros

Técnicas para minimizar el uso de registros en operaciones con objetos.



## Gestión de Pila

Optimización del uso de la pila para llamadas a métodos.



## Rendimiento

Estrategias para reducir la sobrecarga en la gestión de objetos.

# Conclusiones y Desafíos

Concepto	Desafío	Solución
Métodos Virtuales	Resolución dinámica	Tablas virtuales
Herencia	Estructura de memoria	Offset de propiedades
Polimorfismo	Tipo en tiempo de ejecución	RTTI

