



Ejercicio 25

Docente: Jimmy Nataniel Requena Llorentty

Materia: Programación III

Carrera: Ingeniería En Sistemas

Estudiantes: Joaquin Marcos Maita Flores

Santa Cruz – Bolivia

2025

¡ENSAMBLANDO NUESTRO AUTOMOVIL CON SU MOTOR!

```
#include <iostream>
#include <string>

// [1] Clase que representa una parte del automóvil: el Motor
class Motor {
private:
    int caballosDeFuerza; // [2] Atributo: potencia del motor
    bool encendido;      // [3] Atributo: estado del motor (on/off)

public:
    // [4] Constructor con valor por defecto (150 HP)
    Motor(int hp = 150) : caballosDeFuerza(hp), encendido(false) {
        std::cout << " CONSTRUCTOR Motor: Creado motor de " << caballosDeFuerza
        << " HP." << std::endl;
    }

    // [5] Destructor
    ~Motor() {
        std::cout << " DESTRUCTOR Motor: Destruido motor de " <<
        caballosDeFuerza << " HP." << std::endl;
        std::cout << "\033[33m==Joaquin Marcos Maita Flores==\033[0m" <<
        std::endl;
    }

    // [6] Método para encender el motor
    void arrancar() {
        if (!encendido) {
            encendido = true;
            std::cout << " Motor: ¡BRUM! Encendido." << std::endl;
        }
        else {
            std::cout << " Motor: Ya estaba encendido." << std::endl;
        }
    }

    // [7] Método para apagar el motor
    void detener() {
        if (encendido) {
            encendido = false;
            std::cout << " Motor: ...silencio. Apagado." << std::endl;
        }
        else {
            std::cout << " Motor: Ya estaba apagado." << std::endl;
        }
    }
}
```

```
    }  
}  
  
// [8] Método para mostrar estado actual  
void mostrarEstado() const {  
    std::cout << " Estado del Motor: " << (encendido ? "Encendido" :  
"Apagado")  
        << ", HP: " << caballosDeFuerza << std::endl;  
}  
};  
  
// [9] Clase que representa el automóvil completo (contiene un Motor)  
class Automovil {  
private:  
    std::string marca;        // [10] Atributo: marca del auto  
    std::string modelo;      // [11] Atributo: modelo del auto  
    Motor motorInterno;      // [12] Objeto Motor como miembro (COMPOSICIÓN)  
  
public:  
    // [13] Constructor que inicializa todos los miembros  
    Automovil(std::string ma, std::string mo, int hpDelMotor)  
        : marca(ma), modelo(mo), motorInterno(hpDelMotor) {  
        std::cout << "CONSTRUCTOR Automovil: Ensamblado un " << marca << " " <<  
modelo  
            << " con un motor." << std::endl;  
    }  
  
    // [14] Destructor  
    ~Automovil() {  
        std::cout << "DESTRUCTOR Automovil: Desguazando el " << marca << " " <<  
modelo << "." << std::endl;  
        // [15] Nota: motorInterno se destruye automáticamente después  
    }  
  
    // [16] Método para encender el auto (delega al Motor)  
    void encender() {  
        std::cout << modelo << ": Intentando encender..." << std::endl;  
        motorInterno.arrancar();  
    }  
  
    // [17] Método para apagar el auto (delega al Motor)  
    void apagar() {  
        std::cout << modelo << ": Intentando apagar..." << std::endl;  
        motorInterno.detener();  
    }  
}
```

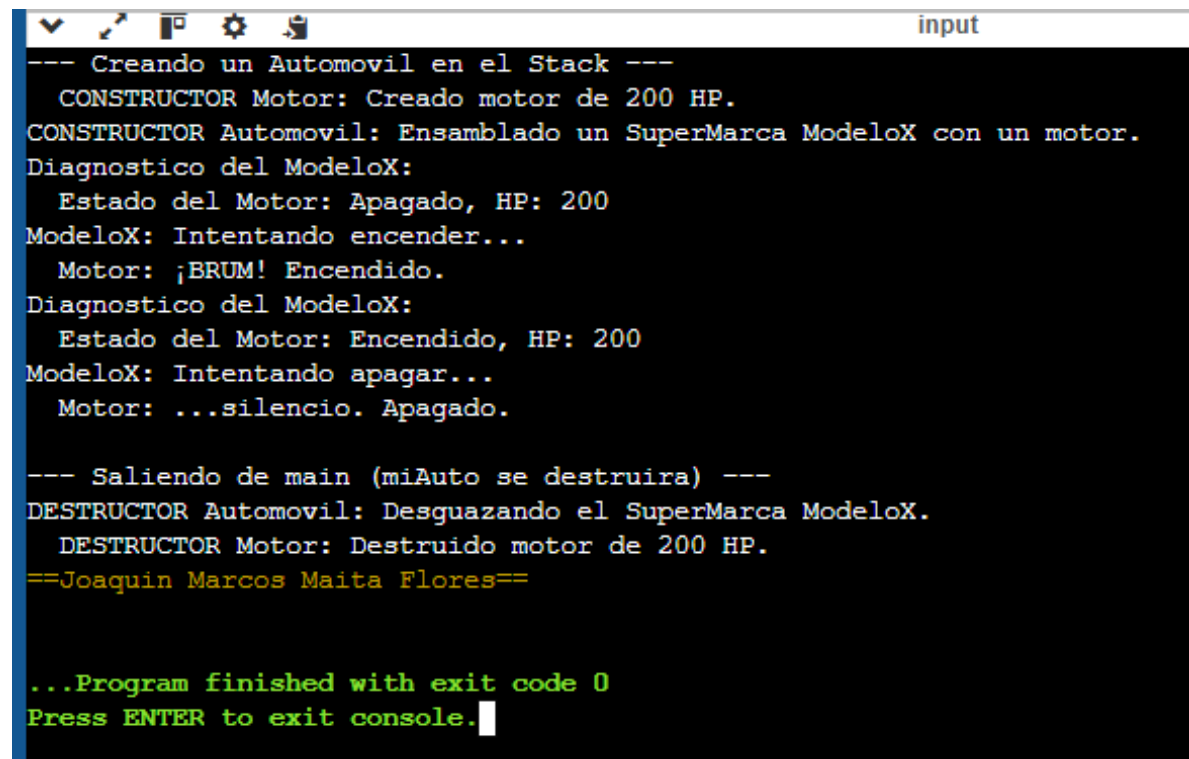
```
// [18] Método para mostrar diagnóstico completo
void verDiagnostico() const {
    std::cout << "Diagnostico del " << modelo << ":" << std::endl;
    motorInterno.mostrarEstado();
}
};

int main() {
    // [19] Creación de objeto Automovil en el stack
    std::cout << "--- Creando un Automovil en el Stack ---" << std::endl;
    Automovil miAuto("SuperMarca", "ModeloX", 200);

    // [20] Operaciones con el auto
    miAuto.verDiagnostico();
    miAuto.encender();
    miAuto.verDiagnostico();
    miAuto.apagar();

    // [21] Fin del ámbito - se destruyen objetos automáticamente
    std::cout << "\n--- Saliendo de main (miAuto se destruió) ---" << std::endl;
    return 0;
}
```

CÓDIGO EJECUTADO



```
input
--- Creando un Automovil en el Stack ---
CONSTRUCTOR Motor: Creado motor de 200 HP.
CONSTRUCTOR Automovil: Ensamblado un SuperMarca ModeloX con un motor.
Diagnostico del ModeloX:
    Estado del Motor: Apagado, HP: 200
ModeloX: Intentando encender...
    Motor: ¡BRUM! Encendido.
Diagnostico del ModeloX:
    Estado del Motor: Encendido, HP: 200
ModeloX: Intentando apagar...
    Motor: ...silencio. Apagado.

--- Saliendo de main (miAuto se destruió) ---
DESTRUCTOR Automovil: Desguazando el SuperMarca ModeloX.
DESTRUCTOR Motor: Destruído motor de 200 HP.
==Joaquin Marcos Maita Flores==

...Program finished with exit code 0
Press ENTER to exit console.
```

Este ejemplo enseña:

1. Composición de clases:

- Automovil contiene un objeto Motor como miembro.
- Demuestra el principio "todo/parte" en POO.

2. Ciclo de vida:

- Al crear un Automovil, primero se construye su Motor.
- Al destruirse el Automovil, primero se ejecuta su destructor y luego el del Motor.

3. Encapsulamiento:

- El Automovil delega las operaciones de motor a su motorInterno.
- Los detalles del motor están ocultos dentro de la clase Automovil.

4. Inicialización:

- Uso de lista de inicialización en constructores.
- Valores por defecto en parámetros.

Anexo

<https://onlinegdb.com/v-nsQosPB>