

# PROGRAMACIÓN DE SISTEMAS

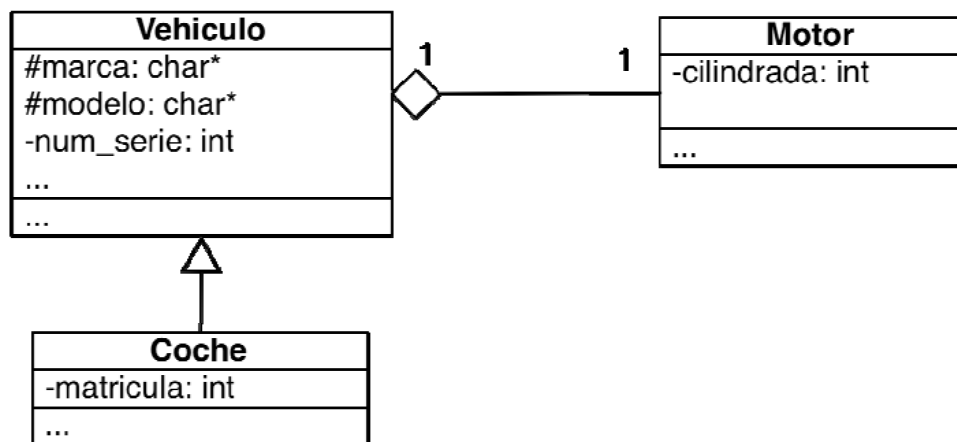
## Ejemplo de Prueba Final

Nombre:

Nº Matrícula:

### Enunciado

Implemente un programa en C++ que, usando las clases descritas en el siguiente diagrama y sus relaciones, imprima en pantalla las matriculas de los coches con cilindrada mayor de 2000. **Añada e implemente cuantos métodos y/o atributos sean necesarios.**



## // DEFINICIONES.H

```
#include <stdio.h>
#include <stdlib.h>
#include <iostream>
```

```
using namespace std;
```

```
class Motor {
```

```
private:
```

```
    int cilindrada;
```

```
public:
```

```
    Motor(int);
```

```
    int getCilindrada();
```

```
    void mostrarCilindrada();
```

```
};
```

```
class Vehiculo {
```

```
private:
```

```
    int numeroSerie;
```

```
    Motor miMotor;
```

```
protected:
```

```
    string marca;
```

```
    string modelo;
```

```
public:
```

```
    Vehiculo(int, string, string, int);
```

```
    //numero serie -> marca -> modelo -> cilindrada
```

```
    Motor getMotor();
```

```
    void mostrarVehiculo();
```

```
};
```

```
class Coche : public Vehiculo {
```

```
private:
```

```
    int matricula;
```

```
public:
```

```
    Coche(int, int, string, string, int);
```

```
    void mostrarMatricula();
```

```
};
```

## // MAIN.CPP

```
#include "Header.h"
```

*// métodos de la clase Motor ↓*

```
Motor::Motor(int _cilindrada) {  
    cilindrada = _cilindrada;  
}  
  
int Motor::getCilindrada() {  
    return cilindrada;  
}  
  
void Motor::mostrarCilindrada() {  
    cout << "La cilindrada de este coche es: " << cilindrada << endl;  
}
```

*// métodos de la clase vehículo ↓*

```
Vehiculo::Vehiculo(int _numSerie, string _marca, string _modelo, int _cilindrada) :  
miMotor(_cilindrada) {  
    numeroSerie = _numSerie;  
    marca = _marca;  
    modelo = _modelo;  
}  
  
Motor Vehiculo::getMotor() {  
    return miMotor;  
}  
  
void Vehiculo::mostrarVehiculo() {  
    cout << "Es el modelo " << modelo;  
    cout << " de la marca " << marca;  
    cout << ".Su numero de serie es: " << numeroSerie << endl;  
}
```

*// métodos de la clase Coche ↓*

```
Coche::Coche(int _matricula, int _numSerie, string _marca, string _modelo, int _cilindrada) :  
Vehiculo(_numSerie, _marca, _modelo, _cilindrada) {  
    matricula = _matricula;  
}  
  
void Coche::mostrarMatricula() {  
    cout << "La matricula de este coche es: " << matricula << endl;  
}
```

```
// función main()

int main() {

    Coche coches[3] = {
        Coche (2698,875,"fiat", "500", 2645),
        Coche (8754,664,"Audi","A1",5950),
        Coche (9874,35,"Seat", "Leon",1500),
    };

    for (int i = 0; i < 3; i++) {

        if (coches[i].getMotor().getCilindrada() > 2000) {

            coches[i].mostrarVehiculo();
            coches[i].mostrarMatricula();
            coches[i].getMotor().mostrarCilindrada();
            cout << "\n";

        }

    }

    system("pause");
    return 0;

}
```

// La salida por pantalla será esta:

```
C:\Users\Jaime\Desktop\Ejem X + v

Es el modelo 500 de la marca fiat .Su numero de serie es: 875
La matricula de este coche es: 2698
La cilindrada de este coche es: 2645

Es el modelo A1 de la marca Audi .Su numero de serie es: 664
La matricula de este coche es: 8754
La cilindrada de este coche es: 5950

Presione una tecla para continuar . . . |
```

**NOTA:** En este ejercicio, he decidido hacer dos únicos archivos, un único .h (con todas las definiciones de las clases), y un único .cpp, en el que defino todos los métodos de las clases, así como la función main.

También podría haber separado las clases en su propio .h y .cpp, el resultado hubiera sido el mismo.