

Herança

Classe Animal.h (classe base/mãe)

```
#include <string>
#include <iostream>
using namespace std;

// Classe base
class Animal {
protected:
    string nome;

public:
    // Construtor
    Animal(string n);

    // Método para emitir som (polimórfico)
    virtual void som();

    // Método para exibir o nome
    void mostraNome();

};
```

Classe Animal.cpp (classe base/ mãe)

```
#include "Animal.h"

// Construtor da classe Animal
Animal::Animal(string n) : nome(n) {}

// Método para emitir som genérico
void Animal::som() {
    cout << nome << " está fazendo barulho" << endl;
}

// Método para exibir o nome do animal
void Animal::mostraNome() {
    cout << "Este é o " << nome << "." << endl;
}
```

Classe Cachorro.h (classe derivada/filha)

```
#include "Animal.h"

// Classe derivada
class Cachorro : public Animal {
public:
    // Construtor para a classe derivada
    Cachorro(string n);
    Cachorro();

    // Método sobrescrito para emitir som específico
    void som() override;

    // Método exclusivo da classe Dog
    void buscar();
};
```

Classe Cachorro.cpp (classe derivada/filha)

```
#include "Cachorro.h"

// Construtor da classe Dog
Cachorro::Cachorro(string n) : Animal(n) {}

// Método sobrescrito para emitir som específico de cachorro (polimorfismo)
void Cachorro::som() {
    cout << nome << " diz: Au!Au!" << endl;
}

// Método exclusivo para buscar a bola
void Cachorro::buscar() {
    cout << nome << " está pegando a bola!" << endl;
}
```

Classe main.cpp

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

#include "Cachorro.h"
#include "Animal.h"
using namespace std;
```

```
int main() {  
    // Criando um objeto da classe base  
    Animal animalGenerico("Animal Genérico");  
    animalGenerico.mostraNome();  
    animalGenerico.som();  
  
    cout << endl;  
  
    // Criando um objeto da classe derivada  
    Cachorro meuCachorro("Buddy");  
    meuCachorro.mostraNome();  
    meuCachorro.som();  
    meuCachorro.buscar();  
  
    return 0;  
}
```