

Семинар #2: Полиморфизм

Полиморфизм

Полиморфизм – это способность функций обрабатывать данные разных типов.

Статический полиморфизм

Динамический полиморфизм

Виртуальные функции

Указатели на базовый класс, хранящие адрес объекта производного класса

Виртуальный деструктор

`override` и `final`

Полиморфизм и умные указатели

Виртуальные функции в конструкторах и деструкторах

Если производный класс вызывает конструктор базового класса (это происходит автоматически в перед вызовом конструктора производного класса) и в конструкторе базового класса вызывается виртуальный метод, то вызовется метод базового класса. Это происходит просто потому что объект производного класса не может вызывать свои методы, так как он ещё не готов к этому.

```
#include <iostream>
struct Alice
{
    Alice()    {say();}
    void func() {say();}
    virtual void say()
    {
        std::cout << "Alice" << std::endl;
    }
};

struct Bob : public Alice
{
    void say() override
    {
        std::cout << "Bob" << std::endl;
    }
};

int main()
{
    Bob b;      // Напечатает Alice
    b.func();   // Напечатает Bob
}
```

Похожая ситуация будет и при вызове виртуального метода в деструкторе:

```
#include <iostream>
struct Alice
{
    virtual void say() {std::cout << "Alice" << std::endl;}
    void func() {say();}
    virtual ~Alice()    {say();}
};

struct Bob : public Alice
{
    void say() override {std::cout << "Bob" << std::endl;};
    ~Bob() {}
};

int main()
{
    Bob b;
    b.func(); // Напечатает Bob
              // Напечатает Alice при уничтожении объекта
}
```

Как программа понимает, какой виртуальный метод вызывать

Размер объектов полиморфных типов. Скрытый указатель на таблицу виртуальных функций.

Абстрактные функции

Чистые виртуальные функции

Абстрактные классы и интерфейсы

Контейнер указателей на базовый класс, хранящих адрес объектов производных классов

pure virtual call и определение чистых виртуальных методов

RTTI и dynamic_cast

Полиморфный тип

dynamic_cast

dynamic_cast от родителя к ребёнку

dynamic_cast в бок

Оператор typeid и класс std::type_info

Реализация механизма виртуальных функций