Федеральное государственное бюджетное образовательное учреждение высшего образования «Московский государственный технический университет имени Н. Э. Баумана (национальный исследовательский университет)

**Лабораторная работа № 3  
Наследование классов в языке C++**

Работу выполнил:

Конов Михаил, ИУ8-24

**ЦЕЛЬ**

Цель работы состоит в овладении навыками разработки программ на языке Си++, использующих возможности наследования классов для решения различных задач.

**УСЛОВИЕ ЗАДАЧИ**  
**ВАРИАНТ 10**

Создать базовый класс «автомобиль». Элементы класса: поле, содержащее наименование модели автомобиля; поле, содержащее значение максимальной скорости (статус доступа *protected*); конструктор для инициализации полей; функция для печати параметров автомобиля. Создать производный класс «автобус». Элементы класса: дополнительно поле, содержащее максимальное число перевозимых пассажиров; конструктор для инициализации полей; переопределенная функция печати параметров автобуса (внутри переопределенной функции должна вызываться функция из базового класса). Создать по 1 объекту каждого из классов. Показать вызов созданных функций. При переопределении функций обеспечить и продемонстрировать два варианта: статический полиморфизм и динамический полиморфизм.

**ТЕКСТ ПРОГРАММЫ**

#include <iostream>

#include <string>

using namespace std;

class Automobile{

protected:

string name;

float velocity;

public:

Automobile(string name, float velocity){

this->name = name;

this->velocity = velocity;

}

virtual void print(){

cout << "\nName: " << name << "\nMaximum velocity: " << velocity;

}

};

class Autobus : public Automobile{

int passengers;

public:

Autobus(string name, float velocity, int passengers):

Automobile(name, velocity){

this->passengers = passengers;

}

void print(){

Automobile::print();

cout << "\npassengers = " << passengers << endl; ;

}

};

int main()

{

setlocale(LC\_ALL, "Russian");

Automobile vazik("УАЗ-31519", 120.0);

Autobus pazik("ПАЗ-3205", 90.0, 41);

Automobile \*aa;

aa = &vazik;

aa -> print();

//динамический полиморфизм. указатель типа базового

//класса Automobile, ссылается на функцию,

//определенную в базовом с модификатором virtual

//и переопределенную в производном классе Autobus

aa = &pazik;

aa -> print();

//статический полиморфизм. указатель типа прозиводного

//класса Autobus, ссылается на функцию, переопределенную в

//классе Autobus

Autobus \*ab;

ab = &pazik;

ab -> print();

system("pause");

return 0;

}

**ВЫВОД**

В ходе лабораторной работы были использованы такие приемы в программировании на языке С++, как создание производных классов, статический и динамический полиморфизм. Изученные способы помогают реализовать решение различных задач, связанных с различными типами данных и объектами.