

## 1. Цель работы

Изучить механизм создания нового класса на основе уже существующего, варианты доступа к элементам базового класса из производного.

## 2. Задание

Согласно варианту №5:

Разработать программу с использованием наследования классов, реализующую классы: железнодорожный вагон(вес порожняка, производитель, год производства); вагон для перевозки автомобилей(кол-во автомобилей, вес автомобиля); цистерна(чем загружена, вес груза). В базовом классе предусмотреть метод вывода параметров вагона. В производных классах предусмотреть методы расчета веса груженого вагона, количества лет эксплуатации вагона

## 3. Листинг программы

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

using namespace std;

// БАЗОВЫЙ ВАГОН

class Carriage
{
protected:
    string name;
    string country_creator;
    string country_arrival;
    int create_year;
    float base_weight;
public:
    void show_info();
    Carriage(string, string, string, int, float);
};

Carriage::Carriage(string name, string country, string country_2, int year = 2000,
float weight = 100)
{
    this->name = name;
    this->country_creator = country;
    this->country_arrival = country_2;
    this->create_year = year;
    this->base_weight = weight;
}

void Carriage::show_info()
{
    time_t now = time(0);
    tm ltm;
    localtime_s(&ltm, &now);

    cout << name << endl;
    cout << "Год создания: " << create_year << endl;
    cout << "Время эксплуатации: " << 1900 + ltm.tm_year - create_year << " лет" <<
endl;
```

```

        cout << "Страна создания: " << country_creator << endl;
        cout << "Пункт назначения: " << country_arrival << endl;
        cout << "Вес без груза: " << base_weight << "кг" << endl;
    }

    // Ж/Д ВАГОН

class RW_Carriage: public Carriage
{
private:
    float cargo_weight;
public:
    void show_info();
    RW_Carriage(string, string, string, int, float, float);
};

void RW_Carriage::show_info()
{
    Carriage::show_info();
    cout << "Вес груза: " << cargo_weight << "кг" << endl;
    cout << "Общий вес: " << cargo_weight + base_weight << "кг" << endl;
    cout << endl;
}

RW_Carriage::RW_Carriage(string name, string country, string country_2, int year =
2000, float weight = 100, float cargo_weight = 1000): Carriage(name, country,
country_2, year, weight)
{
    this->cargo_weight = cargo_weight;
}

// АВТОМОБИЛЬНЫЙ ВАГОН

class Avto_Carriage : public Carriage
{
private:
    float avto_count;
    float avto_weight;
public:
    void show_info();
    Avto_Carriage(string, string, string, int, float, float, float);
};

void Avto_Carriage::show_info()
{
    Carriage::show_info();
    cout << "Количество автомобилей: " << avto_count << endl;
    cout << "Вес автомобиля: " << avto_weight << "кг" << endl;
    cout << "Общий вес: " << base_weight + avto_weight * avto_count << "кг" << endl;
    cout << endl;
}

Avto_Carriage::Avto_Carriage(string name, string country, string country_2, int year
= 2000, float weight = 1000, float avto_count = 10, float avto_weight = 1000):
Carriage(name, country, country_2, year, weight)
{
    this->avto_count = avto_count;
    this->avto_weight = avto_weight;
}

// Цистерна

class Liquid_Carriage : public Carriage
{
private:

```

```

        string liquid;
        float liquid_weight;
public:
    void show_info();
    Liquid_Carriage(string, string, string, int, float, string, float);
};

void Liquid_Carriage::show_info()
{
    Carriage::show_info();
    cout << "Жидкость: " << liquid << endl;
    cout << "Вес жидкости: " << liquid_weight << endl;
    cout << "Общий вес: " << base_weight + liquid_weight << "кг" << endl;
    cout << endl;
}

Liquid_Carriage::Liquid_Carriage(string name, string country, string country_2, int
year = 2000, float weight = 1000, string liquid = "Вода", float liquid_weight =
1000) : Carriage(name, country, country_2, year, weight)
{
    this->liquid = liquid;
    this->liquid_weight = liquid_weight;
}

int main()
{
    setlocale(LC_ALL, "RUS");
    RW_Carriage RWcargo("Железнодорожный вагон", "Россия", "Китай", 2005, 1000,
5000);
    Avto_Carriage AVTOcargo("Автомобильный вагон", "Германия", "Китай", 2019, 1100,
20, 1300);
    Liquid_Carriage LD_Carriage("Вагон Цистерна", "Азербайджан", "Казахстан", 2021,
900, "Нефть", 10000);

    RWcargo.show_info();
    AVTOcargo.show_info();
    LD_Carriage.show_info();
}

```

#### 4. Пример работы программы

```
Железнодорожный вагон
Год создания: 2005
Время эксплуатации: 19 лет
Страна создания: Россия
Пункт назначения: Китай
Вес без груза: 1000кг
Вес груза: 5000кг
Общий вес: 6000кг

Автомобильный вагон
Год создания: 2019
Время эксплуатации: 5 лет
Страна создания: Германия
Пункт назначения: Китай
Вес без груза: 1100кг
Количество автомобилей: 20
Вес автомобиля: 1300кг
Общий вес: 27100кг

Вагон Цистерна
Год создания: 2021
Время эксплуатации: 3 лет
Страна создания: Азербайджан
Пункт назначения: Казахстан
Вес без груза: 900кг
Жидкость: Нефть
Вес жидкости: 10000
Общий вес: 10900кг
```

Рисунок 1 – Пример работы

## 5. Выводы

В процессе лабораторной работы были изучены принципы наследования классов, варианты доступа к данным базового класса через производный.