

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Воронежский государственный технический университет»
(ФГБОУ ВО «ВГТУ», ВГТУ)

ОТЧЕТ ПО ПРЕДМЕТУ СРЕДЫ ВИЗУАЛЬНОГО ПРОГРАММИРОВАНИЯ
ЛАБОРАТОРНАЯ 2.

Обучающийся _____ Тихоненко Дмитрий Анатольевич
(ФИО обучающегося)

Группа БИВТ-222

Наименование предприятия ВГТУ

Обучающийся	_____	Д.А.Тихоненко
	(подпись)	

Руководитель по практической подготовке	_____	А.С. Троценко
	(подпись)	

Users > macbookpro13retina2015 > Desktop > C laba.h > Vehicle

```
1  #include <iostream>
2  #include <string>
3  using namespace std;
4
5
6  class Vehicle
7  {
8  protected:
9      int mileage; // пробег
10     double fuel; // количество топлива в баке
11     const double tankCapacity; // вместимость бака
12     const double fuelConsumption; // расход топлива
13 public:
14     Vehicle(double capacity, double consumption) : mileage(0), fuel(0), tankCapacity(capacity), fuelConsumption(consumption) {}
15     void drive(int kilometers)
16     {
17         double fuelNeeded = (static_cast<double>(kilometers) / 100.0) * fuelConsumption;
18         if (fuel >= fuelNeeded)
19         {
20             mileage += kilometers;
21             fuel -= fuelNeeded;
22             cout << "Проехали " << kilometers << " км. Осталось топлива: " << fuel << " литров." << endl;
23         }
24         else
25         {
26             cout << "Недостаточно топлива для поездки на " << kilometers << " км." << endl;
27         }
28     }
29     void refuel()
30     {
31         fuel = tankCapacity;
32         cout << "Бак заправлен до полного. Теперь у вас " << fuel << " литров топлива." << endl;
33     }
34     void printStatus()
35     {
36         cout << "Пробег: " << mileage << " км. Осталось топлива: " << fuel << " литров." << endl;
37     }
38 }
```

Users > macbookpro13retina2015 > Desktop > C laba.h > Vehicle

```
31     fuel = tankCapacity;
32     cout << "Бак заправлен до полного. Теперь у вас " << fuel << " литров топлива." << endl;
33 }
34 void printStatus()
35 {
36     cout << "Пробег: " << mileage << " км. Осталось топлива: " << fuel << " литров." << endl;
37 }
38 };
39
40 class Sedan : public Vehicle
41 {
42 public:
43     Sedan() : Vehicle(50.0, 6.5) {}
44 };
45
46 class Suv : public Vehicle
47 {
48 public:
49     Suv() : Vehicle(70.0, 8.0) {}
50 };
51
52 class Bus : public Vehicle
53 {
54 public:
55     Bus() : Vehicle(200.0, 15.0) {}
56 };
57
```

```

Users > macbookpro13retina2015 > Desktop > laba2.cpp > main()
1  #include "laba.h"
2  int main() {
3      setlocale(0, "");
4      Vehicle* vehicles[4] = { new Sedan(), new Suv(), new Bus()};
5      for (int i = 0; i < 4; ++i) {
6          Vehicle* vehicle = vehicles[i];
7          vehicle->refuel();
8          vehicle->printStatus();
9          vehicle->drive(120);
10         vehicle->printStatus();
11         vehicle->drive(420);
12         vehicle->printStatus();
13         vehicle->drive(300);
14         vehicle->printStatus();
15         vehicle->refuel();
16         vehicle->drive(300);
17         vehicle->printStatus();
18         delete vehicle;
19         cout << endl;
20     }
21     return 0;}
22

```

Пояснение задачи:

В начале кода мы включаем заголовочные файлы `<iostream>` и `<string>`, а также используем директиву `using namespace std`. Это обеспечивает использование стандартного пространства имен `std`, чтобы не писать `std::` перед стандартными именами, такими как `cout`, `endl` и т.д.

Затем мы определяем класс `Vehicle` с защищёнными членами данных: `mileage` (пробег), `fuel` (количество топлива), `tankCapacity` (ёмкость бака) и `fuelConsumption` (расход топлива). В конструкторе этого класса инициализируются параметры топливного бака и расхода топлива. Есть методы для езды (`drive`), заправки (`refuel`) и вывода статуса (`printStatus`).

Далее у нас есть три производных класса: `Sedan`, `Suv`, и `Bus`, каждый из которых представляет различные типы транспортных средств. Каждый из этих классов инициализирует параметры базового класса `Vehicle` своими уникальными значениями в конструкторе.

В функции `main` создаётся массив указателей на объекты класса `Vehicle` (или его производных). Затем в цикле для каждого объекта выполняются следующие действия:

- Заправка транспортного средства.
- Вывод текущего статуса (пробег и количество топлива).
- Проезд 120 км.
- Вывод статуса после поездки.
- Проезд 420 км.
- Вывод статуса после второй поездки.
- Проезд еще 300 км.
- Вывод статуса после третьей поездки.
- Заправка снова.
- Проезд еще 300 км.
- Вывод статуса после четвёртой поездки.

После завершения работы с каждым транспортным средством, происходит удаление объекта (освобождение памяти с помощью delete).

Таким образом, этот код моделирует использование различных типов транспортных средств, их заправку, поездки и вывод статуса.