Министерство науки и высшего образования Российской Федерации  
Пензенский государственный университет  
Кафедра вычислительная техника

**ОТЧËТ**  
по лабораторной работе №1  
по курсу «Объектно-ориентированное программирование»  
на тему «Основы работы с классами и объектами на C++»

Выполнили студенты группы 22ВВП1:

Ипполитов И. Д.

Хоссейни Нежад С. А. С. М.

Приняли:  
Евсеева Ю.И.  
Гудков А. А.

Пенза 2024

**Название**

Основы работы с классами и объектами на C++

**Цель работы**

Изучить основы работы с классами и объектами на языке C++

**Лабораторное задание**

Разработать классы для описанных ниже объектов. Включить в класс конструктор по умолчанию, конструктор с параметрами, конструктор копирования и деструктор. Поля класса определить с модификатором доступа private, для доступа к ним реализовать методы Set- и Get- с модификатором доступа public.

**Train**: Пункт назначения, Номер поезда, Время отправления, Число общих мест, Купейных, Плацкартных. Создать массив объек- тов. Вывести:

а) список поездов, следующих до заданного пункта назначения;

б) список поездов, следующих до заданного пункта назначения и отправляющихся после заданного часа;

в) список поездов, отправляющихся до заданного пункта назначе- ния и имеющих общие места

**Метод решения задачи**

Разработали классы согласно заданию, включили конструкторы, деструкторы, геттеры и сеттеры

**Листинг**

#include <iostream>

#include <string>

#include <windows.h>

#include <vector>

using namespace std;

class Train {

private:

string destination;

int number;

int time;

int mainCount;

int kupeCount;

int plackartCount;

public:

Train(string destination, int number, int time, int mainCount, int kupeCount, int plackartCount)

{

this->destination = destination;

this->number = number;

this->time = time;

this->mainCount = mainCount;

this->kupeCount = kupeCount;

this->plackartCount = plackartCount;

}

Train(const Train& t)

{

this->destination = t.destination;

this->number = t.number;

this->time = t.time;

this->mainCount = t.mainCount;

this->kupeCount = t.kupeCount;

this->plackartCount = t.plackartCount;

}

~Train()

{

}

void show() {

cout << endl;

cout << "Пункт назначения: " << destination << endl;

cout << "Номер: " << number << endl;

cout << "Время: " << time << endl;

cout << "Общее количество мест: " << mainCount << endl;

cout << "Мест в купе: " << kupeCount << endl;

cout << "Мест в плацкарте: " << plackartCount << endl;

}

string getDestination() { return destination; }

int getTime() { return time; }

int getMainCount() { return mainCount; }

int getNumber() { return number; }

int getKupeCount() { return kupeCount; }

int getPupeCount() { return plackartCount; }

string setDestination(string destination) {

this->destination = destination;

}

int setTime(int time) {

this->time = time;

}

int setMainCount(int mainCount) {

this->mainCount = mainCount;

}

int setNumber(int number) {

this->number = number;

}

int setKupeCount(int kupeCount) {

this->kupeCount = kupeCount;

}

int setPupeCount(int plackartCount) {

this->plackartCount = plackartCount;

}

};

int main()

{

setlocale(LC\_ALL, "rus");

SetConsoleOutputCP(1251);

SetConsoleCP(1251);

string destination;

int number;

int time;

int mainCount;

int kupeCount;

int plackartCount;

vector<Train> v1; // массив поездов

for (size\_t i = 0; i < 5; i++)

{

cout << "Введите данные о поезде " << i + 1<< endl;

cout << "Пункт назначения: "; cin >> destination;

cout << "Номер: "; cin >> number;

cout << "Время: "; cin >> time;

cout << "Общее количество мест: "; cin >> mainCount;

cout << "Мест в купе: "; cin >> kupeCount;

cout << "Мест в плацкарте: "; cin >> plackartCount;

cout << endl;

Train t{ destination, number, time, mainCount, kupeCount, plackartCount };

v1.push\_back(t);

}

cout << "Введите искомый пункт назначения: "; cin >> destination;

cout << "Введите время: "; cin >> time;

cout << "Введите количество мест: "; cin >> mainCount;

cout << "Поезда, следующие до заданного пункта назначения:" << endl;

for (size\_t i = 0; i < v1.size(); i++)

{

if (v1[i].getDestination() == destination)

{

cout << v1[i].getNumber() << endl;

}

}

cout << endl;

cout << "Поезда, следующие до заданного пункта назначения с учетом времени:" << endl;

for (size\_t i = 0; i < v1.size(); i++)

{

if (v1[i].getDestination() == destination && v1[i].getTime() > time)

{

cout << v1[i].getNumber() << endl;

}

}

cout << endl;

cout << "Поезда, следующие до заданного пункта назначения с учетом мест:" << endl;

for (size\_t i = 0; i < v1.size(); i++)

{

if (v1[i].getDestination() == destination && v1[i].getMainCount() == mainCount)

{

cout << v1[i].getNumber() << endl;

}

}

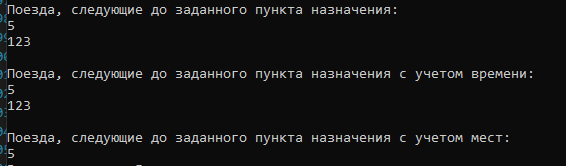
cout << "Размер массива: " << v1.size();

}

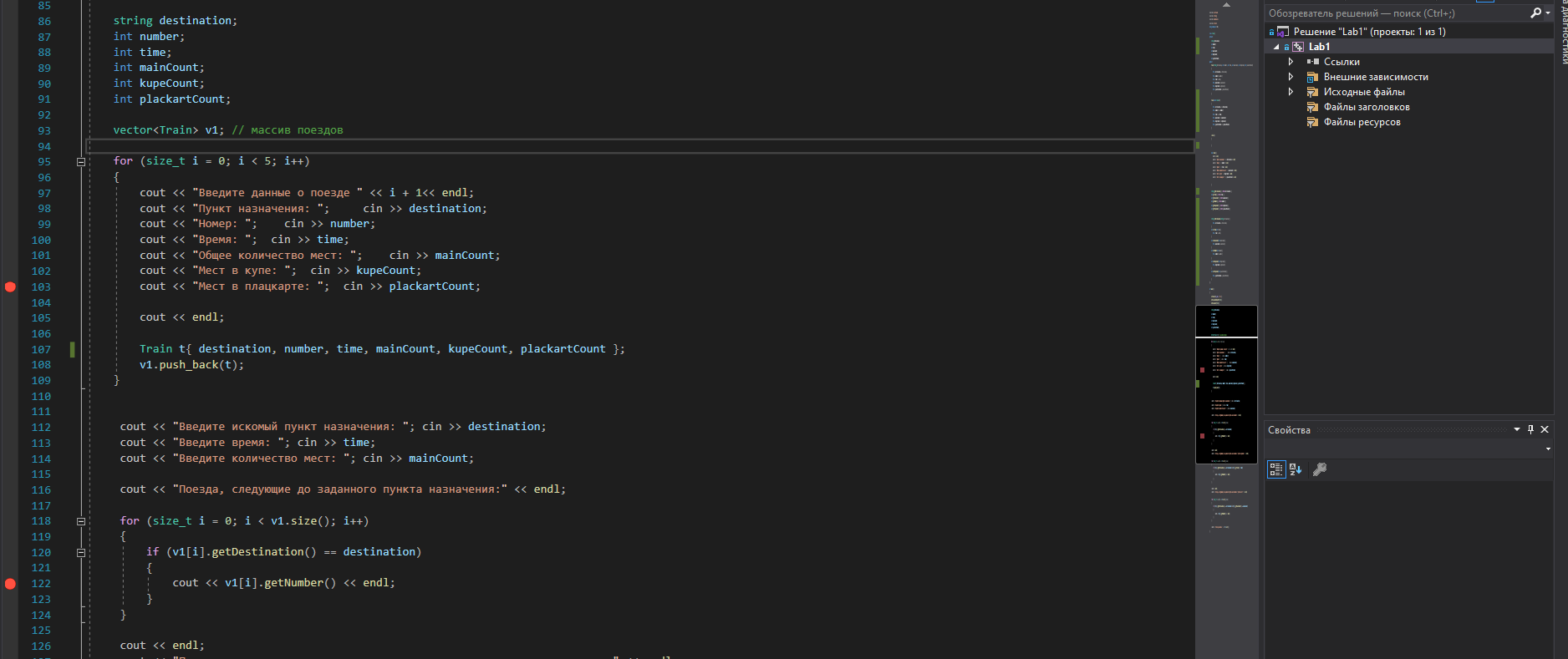
**Пояснительный текст к программе**

Создается несколько продуктов с данными, введенными с клавиатуры. Далее ищем поезда, которые следуют в указанный город. Затем время отправлений которых больше заданного.

**Результат работы программы**

****

**Протокол трассировки программы**

****

**Вывод**

Мы изучили основы работы с классами и объектами на языке C++