# **Прикладное программирование**

Практика 2

Конструктор класса

Каждую программу сохранять в отдельном файле для каждого упражнения,

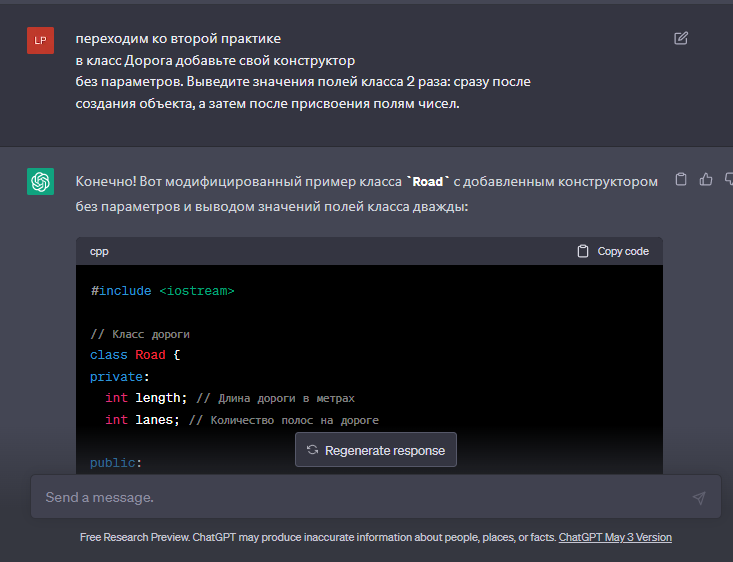
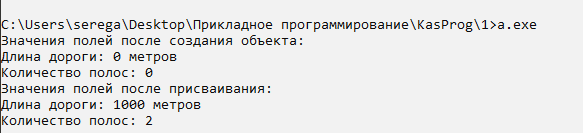
например prog02-1, prog02-2 и т.д, где "02" - номер практического

занятия, 1, 2 и т.д. - номер задания.

1. В пример 1 из лекции в класс Дорога добавьте свой конструктор

без параметров. Выведите значения полей класса 2 раза: сразу после

создания объекта, а затем после присваивания полям чисел.

  
  
#include <iostream>

// Класс дороги

class Road {

private:

int length; // Длина дороги в метрах

int lanes; // Количество полос на дороге

public:

// Конструктор без параметров

Road() {

length = 0;

lanes = 0;

}

// Метод для установки значений полей

void setValues(int roadLength, int roadLanes) {

length = roadLength;

lanes = roadLanes;

}

// Метод для вывода значений полей

void printValues() {

std::cout << "Длина дороги: " << length << " метров" << std::endl;

std::cout << "Количество полос: " << lanes << std::endl;

}

};

int main() {

// Создаем объект дороги с использованием конструктора без параметров

Road road1;

std::cout << "Значения полей после создания объекта:" << std::endl;

road1.printValues();

// Присваиваем значения полям

road1.setValues(1000, 2);

// Выводим значения полей после присваивания

std::cout << "Значения полей после присваивания:" << std::endl;

road1.printValues();

return 0;

}

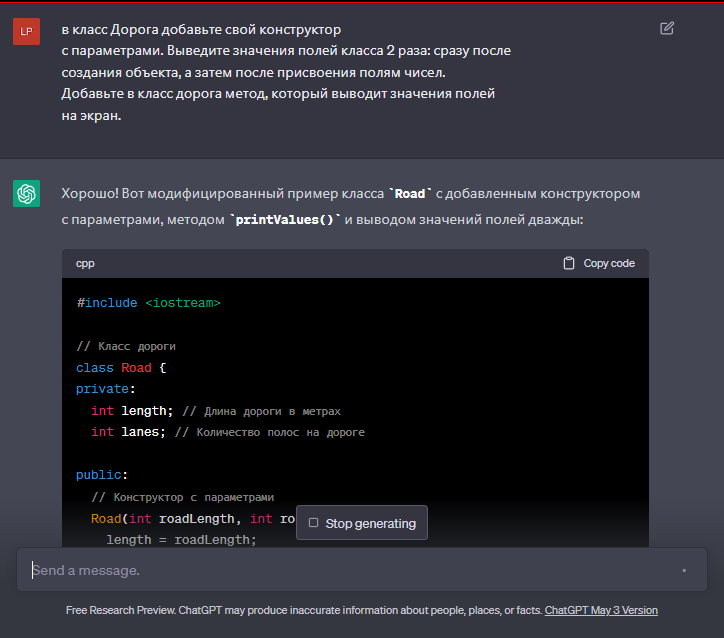
2. В пример 1 из лекции в класс Дорога добавьте свой конструктор

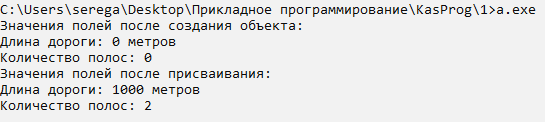
с параметрами. Выведите значения полей класса 2 раза: сразу после

создания объекта, а затем после присвоения полям чисел.

Добавьте в класс дорога метод, который выводит значения полей

на экран.



  
#include <iostream>

// Класс дороги

class Road {

private:

int length; // Длина дороги в метрах

int lanes; // Количество полос на дороге

public:

// Конструктор с параметрами

Road(int roadLength, int roadLanes) {

length = roadLength;

lanes = roadLanes;

}

// Метод для вывода значений полей

void printValues() {

std::cout << "Длина дороги: " << length << " метров" << std::endl;

std::cout << "Количество полос: " << lanes << std::endl;

}

};

int main() {

// Создаем объект дороги с использованием конструктора с параметрами

Road road1(0, 0);

std::cout << "Значения полей после создания объекта:" << std::endl;

road1.printValues();

// Присваиваем значения полям

road1 = Road(1000, 2);

// Выводим значения полей после присваивания

std::cout << "Значения полей после присваивания:" << std::endl;

road1.printValues();

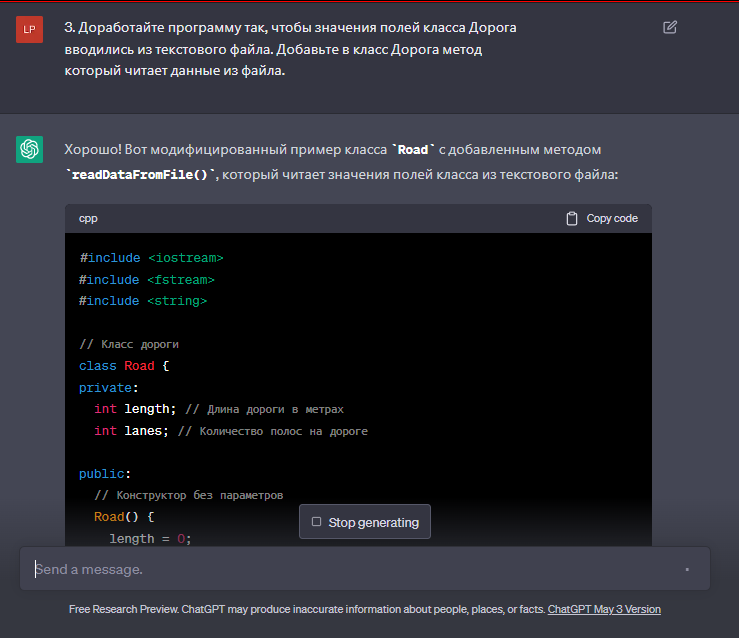
return 0;

}

3. Доработайте программу так, чтобы значения полей класса Дорога

вводились из текстового файла. Добавьте в класс Дорога метод

который читает данные из файла.





#include <iostream>

#include <fstream>

#include <string>

// Класс дороги

class Road {

private:

int length; // Длина дороги в метрах

int lanes; // Количество полос на дороге

public:

// Конструктор без параметров

Road() {

length = 0;

lanes = 0;

}

// Метод для чтения данных из файла

void readDataFromFile(const std::string& fileName) {

std::ifstream inputFile(fileName);

if (inputFile.is\_open()) {

inputFile >> length >> lanes;

inputFile.close();

} else {

std::cout << "Не удалось открыть файл: " << fileName << std::endl;

}

}

// Метод для вывода значений полей

void printValues() {

std::cout << "Длина дороги: " << length << " метров" << std::endl;

std::cout << "Количество полос: " << lanes << std::endl;

}

};

int main() {

Road road;

// Чтение данных из файла

road.readDataFromFile("input.txt");

// Вывод значений полей

road.printValues();

return 0;

}

4. Создайте класс Книга, в котором использованы 2 вида конструкора:

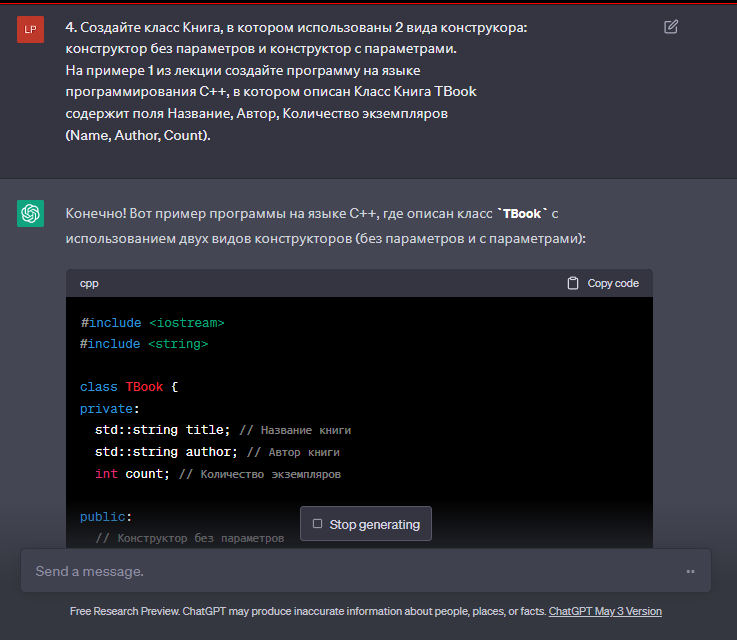
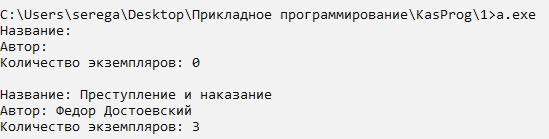
конструктор без параметров и конструктор с параметрами.

На примере 1 из лекции создайте программу на языке

программирования C++, в котором описан Класс Книга TBook

содержит поля Название, Автор, Количество экземпляров

(Name, Author, Count).

#include <iostream>

#include <string>

class TBook {

private:

std::string title; // Название книги

std::string author; // Автор книги

int count; // Количество экземпляров

public:

// Конструктор без параметров

TBook() {

title = "";

author = "";

count = 0;

}

// Конструктор с параметрами

TBook(const std::string& bookTitle, const std::string& bookAuthor, int bookCount) {

title = bookTitle;

author = bookAuthor;

count = bookCount;

}

// Метод для вывода информации о книге

void printInfo() {

std::cout << "Название: " << title << std::endl;

std::cout << "Автор: " << author << std::endl;

std::cout << "Количество экземпляров: " << count << std::endl;

}

};

int main() {

// Создание объекта книги с использованием конструктора без параметров

TBook book1;

book1.printInfo();

std::cout << std::endl;

// Создание объекта книги с использованием конструктора с параметрами

TBook book2("Преступление и наказание", "Федор Достоевский", 3);

book2.printInfo();

return 0;

}