

**МИНОБРНАУКИ РОССИИ**

**Федеральное государственное бюджетное  
образовательное учреждение высшего образования  
«Тульский государственный университет»**

**Институт прикладной математики и компьютерных наук  
Кафедра «Информационная безопасность»**

Утверждено на заседании кафедры  
«Информационная безопасность»  
« » \_\_\_\_\_ 202\_\_ г., протокол № \_\_\_\_

Заведующий кафедрой  
\_\_\_\_\_ А.А. Сычугов

**МЕТОДИЧЕСКИЕ УКАЗАНИЯ  
по выполнению курсовой работы  
по дисциплине (модулю)  
«Языки программирования»**

**основной профессиональной образовательной программы  
высшего образования – программы специалитета**

по специальности:

10.05.03 Информационная безопасность автоматизированных систем  
с профилем:

Разработка автоматизированных систем в защищенном исполнении

Форма обучения: очная

Идентификационный номер образовательной программы: 100503-01-21

Тула 2021 год

**ЛИСТ СОГЛАСОВАНИЯ**  
**методических указаний к курсовой работе по дисциплине (модулю)**

**Разработчик:**

Басалова Г.В., к.т.н., доцент

---

*(ФИО, должность, ученая степень, ученое звание)*

---

*(подпись)*

# СОДЕРЖАНИЕ

<b>ВВЕДЕНИЕ .....</b>	<b>4</b>
<b>1. ЦЕЛИ И ЗАДАЧИ ВЫПОЛНЕНИЯ КУРСОВОЙ РАБОТЫ.....</b>	<b>4</b>
<b>2. ОСНОВНЫЕ ТРЕБОВАНИЯ К КУРСОВОЙ РАБОТЕ.....</b>	<b>4</b>
2.1. ТЕМАТИКА КУРСОВОЙ РАБОТЫ .....	4
2.2. ЗАДАНИЕ И ИСХОДНЫЕ ДАННЫЕ К КУРСОВОЙ РАБОТЕ.....	4
2.3. ОБЪЕМ КУРСОВОЙ РАБОТЫ .....	5
2.4. ВЫПОЛНЕНИЕ КУРСОВОЙ РАБОТЫ .....	5
2.5. ЗАЩИТА КУРСОВОЙ РАБОТЫ .....	6
<b>3. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ К ВЫПОЛНЕНИЮ КУРСОВОЙ РАБОТЫ</b>	<b>6</b>
3.1. ПЛАН ПОСТРОЕНИЯ И СОДЕРЖАНИЕ РАЗДЕЛОВ ПОЯСНИТЕЛЬНОЙ ЗАПИСКИ К КУРСОВОЙ РАБОТЕ.....	6
3.2. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ВЫПОЛНЕНИЮ ОТДЕЛЬНЫХ РАЗДЕЛОВ КУРСОВОЙ РАБОТЫ .....	6
3.3. ОФОРМЛЕНИЕ ПОЯСНИТЕЛЬНОЙ ЗАПИСКИ КУРСОВОЙ РАБОТЫ .....	7
<b>4. БИБЛИОГРАФИЧЕСКИЙ СПИСОК.....</b>	<b>9</b>
<b>ПРИЛОЖЕНИЕ.....</b>	<b>10</b>

# ВВЕДЕНИЕ

Целями освоения дисциплины является теоретическая и практическая подготовка специалистов к деятельности, связанной с проектированием и эксплуатацией систем защиты информации автоматизированных систем.

Задачами освоения дисциплины являются:

- получение теоретических знаний и практических навыков проектирования приложений на различных языках программирования;
- освоение принципов тестирования и отладки приложений;
- приобретение навыков использования современных технологий разработки алгоритмов и программ

## 1. ЦЕЛИ И ЗАДАЧИ ВЫПОЛНЕНИЯ КУРСОВОЙ РАБОТЫ

Курсовая работа является важным этапом изучения дисциплины "Языки программирования". Ее выполнение должно способствовать достижению следующих целей: расширение и закрепление знаний по курсу, развитие умения самостоятельно работать с источниками специализированной информации, получение необходимого опыта в проектировании алгоритмов и разработки прикладных решений с помощью различного программного обеспечения и изученных технологий.

## 2. ОСНОВНЫЕ ТРЕБОВАНИЯ К КУРСОВОЙ РАБОТЕ

### 2.1. ТЕМАТИКА КУРСОВОЙ РАБОТЫ

Тематика курсовой работы соответствует профилю направления и отражает актуальные потребности теории и практики использования алгоритмов и языков программирования при разработке различных автоматизированных систем. Реализация может происходить с использованием языков программирования C++, C#, Pascal, Python и др. Темы курсовых работ могут отражать следующие направления исследований и разработок:

1. Проектирование консольного приложения
2. Проектирование приложения с оконным интерфейсом
3. Исследование алгоритмов обработки данных
4. Изучение и сравнительный анализ алгоритмов обработки данных

### 2.2. ЗАДАНИЕ И ИСХОДНЫЕ ДАННЫЕ К КУРСОВОЙ РАБОТЕ

Задание и исходные данные на курсовую работу выдаются руководителем (преподавателем) на специальном типовом бланке (см. приложение). Отдельные данные могут быть выбраны студентом самостоятельно и согласованы

с руководителем. В задании указываются:

- тема работы;
- исходные данные к работе;
- рекомендуемая литература;
- сроки выполнения и защиты проекта.

### **2.3. ОБЪЕМ КУРСОВОЙ РАБОТЫ**

Курсовая работа состоит из пояснительной записки, задания и рецензии. Рекомендуемый объем пояснительной записки – от 25 страниц (без учета приложений). Увеличение объема (не более 40 страниц) может потребоваться для полного раскрытия темы курсовой работы.

### **2.4. ВЫПОЛНЕНИЕ КУРСОВОЙ РАБОТЫ**

Работа над курсовой работой выполняется по графику, определяемому руководителем. В целях ее планомерного выполнения рекомендуется следующий график работы.

<i>Неделя</i>	<i>Содержание работ</i>	<i>Результаты работы</i>
1 -2	Получение и ознакомление с заданием.	Заполненный бланк задания.
3–4	Изучение литературы и других исходных материалов.	Обзор литературы. Выбор основных источников
5–6	Выбор конкретных алгоритмов и способов реализации	Алгоритмы
7-10	Реализация алгоритмов	Программный код КР
11 - 13	Тестирование программной части работы	Сформированный набор тестов
14–15	Оформление пояснительной записки и сдача на проверку.	Пояснительная записка.
16	Защита курсовой работы.	

По всем вопросам, возникающим при выполнении курсовой работы, необходимо обращаться к руководителю, назначаемому кафедрой. Все результаты работы должны предъявляться руководителю для проверки в соответствии с графиком выполнения работы.

Руководитель также проверяет пояснительную записку. Все ошибки, недоработанные места указываются и разъясняются студенту. Если работа удовлетворяет требованиям, то руководитель подписывает пояснительную записку, тем самым допуская его к защите.

## **2.5. ЗАЩИТА КУРСОВОЙ РАБОТЫ**

Курсовая работа представляется на кафедру для проверки за неделю до ее защиты. При положительной оценке руководителем студент допускается к защите работы перед комиссией.

Защита – форма проверки выполненной работы. Курсовая работа защищается публично в присутствии студентов перед комиссией, назначаемой заведующим кафедрой. Руководитель работы является членом комиссии. При защите работы сначала студент выступает с сообщением продолжительностью 8–10 минут по существу работы. Затем по докладу и содержанию пояснительной записки студенту задаются вопросы членами комиссии, на которые он должен ответить.

Курсовая работа оценивается по 100-балльной системе с учетом:

- обоснованности объема (соответствия заданию) и качества выполнения курсовой работы;
- степени самостоятельности при выполнении работы;
- качества оформления пояснительной записки и соответствия их требованиям данных методических указаний;
- качества защиты и правильности ответов на вопросы.

Студент, не представивший в срок курсовую работу или не защитивший ее по неуважительной причине, считается имеющим академическую задолженность.

В случае получения неудовлетворительной оценки студенту выдается новое задание.

## **3. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ К ВЫПОЛНЕНИЮ КУРСОВОЙ РАБОТЫ**

### **3.1. ПЛАН ПОСТРОЕНИЯ И СОДЕРЖАНИЕ РАЗДЕЛОВ ПОЯСНИТЕЛЬНОЙ ЗАПИСКИ К КУРСОВОЙ РАБОТЕ**

Пояснительная записка должна иметь структуру: титульный лист, задание, содержание, введение, основную часть, заключение, список использованных источников, приложения.

К пояснительной записке прилагается незаполненный бланк рецензии (см. приложение).

### **3.2. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ВЫПОЛНЕНИЮ ОТДЕЛЬНЫХ РАЗДЕЛОВ КУРСОВОЙ РАБОТЫ**

Форма титульного листа приведена в приложении. Задание, оформленное в соответствии с требованиями, помещается в пояснительной записке на втором листе. Страница с заданием имеет номер 2. Задание должно быть подписано руководителем работы и студентом.

Содержание включает наименования всех разделов, подразделов и пунктов с указанием номеров страниц.

Примерное содержание курсовой работы на тему «Изучение и сравнительный анализ алгоритмов обработки данных» имеет вид:

Введение

1 Постановка задачи

2 Описание исследуемого (-ых) алгоритма (-ов) (Теоретическая часть)

3 Возможности применения алгоритма

4 Решение задач с использованием алгоритма

Заключение

Список литературы

Приложения

Введение должно отражать историю рассматриваемого вопроса или метода, его теоретическую и практическую значимость (объем введения – ок. 1 страницы).

Основная часть содержит постановку задачи, изложение теоретических положений с необходимыми выводами, варианты реализации алгоритма для решения задач, описание разработанных задач для демонстрации применения алгоритма, обоснование и описание разработанных тестов для проверки работоспособности решений.

Пример оформления основной части КР приведен в приложении. Если задание на КР включает составление задач по изучаемым алгоритмам, то описание задач должно включать следующие части: условие, описание входных и выходных данных, алгоритм решения, решение на одном или двух языках программирования, примеры входных и выходных данных, описание групп тестов для проверки работоспособности решения.

Заключение должно содержать краткие выводы по работе, оценку полученных результатов. В нем отмечается, какие новые знания и навыки получены студентом при выполнении курсовой работы, указываются перспективы развития, мероприятия по совершенствованию разработанных алгоритмов, компонентов и т.д.

В приложения включаются таблицы с исходными данными, тексты программ, громоздкие результаты работы программы или результаты, не представленные в разделе, схемы алгоритмов.

### **3.3 ОФОРМЛЕНИЕ ПОЯСНИТЕЛЬНОЙ ЗАПИСКИ КУРСОВОЙ РАБОТЫ**

Записка оформляется на листах формата А4.

Для основного текста рекомендуется использовать тип шрифта Times New Roman с размер шрифта 14 пт. Переносы слов в заголовках не допускаются. Каждый раздел следует начинать с новой страницы. Расстояние между заголовком и текстом должно быть 10-15 мм.

Разделы должны иметь порядковую нумерацию в пределах всей записки и обозначаться арабскими цифрами без точки в конце. Точка не ставится и в конце всего заголовка. Введение и заключение не нумеруются. Подразделы нумеруют арабскими цифрами в пределах каждого раздела. Номер подраздела состоит из номера раздела и подраздела, разделенных точкой. Например, «2.3» – подраздел 3 раздела 2. Пункты нумеруют в пределах подраздела, например «1.1.2» — пункт 2 подраздела 1 раздела 1.

Иллюстрации, кроме таблиц, обозначаются снизу словом "Рис." и нумеруются последовательно арабскими цифрами в пределах раздела (при небольшом числе рисунков допускается сквозная нумерация по всей работе). Номер иллюстрации должен состоять из номера раздела и порядкового номера иллюстрации, например, «Рис. 1.2» – рисунок 2 раздела 1. Если в записке приведена одна иллюстрация, то ее не нумеруют и слово "Рис." не пишут. Иллюстрация должна иметь наименование. При необходимости к иллюстрации приводят пояснительные надписи (подрисуночный текст). Наименование иллюстрации помещают над ней. Пример подписи под рисунком: «Рис. 1.2 Структурная схема программы». И иллюстрация, и подпись под ней выравниваются по центру.

Цифровой и структурированный материал рекомендуется оформлять в виде таблиц. Таблицы нумеруют в пределах раздела. Каждая таблица должна иметь заголовок, который помещается над самой таблицей. Заголовок и слово «Таблица» пишут с прописной буквы. В правом верхнем углу таблицы над соответствующим заголовком помещают надпись, например, «Таблица 1.2» – таблица 2 раздела 1. Если в записке одна таблица, то ее не нумеруют и слово "Таблица" не пишут.

Формулы в записке нумеруют арабскими цифрами в пределах раздела (при небольшом количестве – до 10 – в пределах всей работы). В первом случае номер формулы состоит из номера раздела и порядкового номера формулы в разделе. Номер помещают в правой стороне страницы на уровне формулы в круглых скобках, например, (3.1) – формула 1 раздела 3.

Уравнения и формулы отделяются от текста по вертикали одинарным межстрочным интервалом. Если соотношение не уместится в одну строку, то оно должно быть перенесено после математических знаков. Формулы конструируются встроенным построителем формул текстового редактора или внешним приложением (например, MS Equation, MathType). Базовый размер шрифта в формулах должен быть таким же, что и размер шрифта основного текста.

Ссылки в тексте на источник указывают порядковым номером по списку источников, выделенным квадратными скобками, например, «в соответствии с [2]». На все источники в списке литературы (возможно, кроме ГОСТ) в работе должны быть ссылки в виде [2], [2, 3], [2-5]. Ссылки на формулы указывают порядковым номером формулы в круглых скобках, например, «в формуле (1.2)». Ссылки на иллюстрации указывают порядковым номером иллюстрации, например, «рис. 1.2». На все таблицы тоже должны быть ссылки в тексте. Слово «таблица» пишется полностью, если таблица не имеет номера, и сокращенно, если имеет номер, например, «табл. 1.2.». В повторных ссылках на таблицы и иллюстрации следует сокращенно указывать слово «смотри», например, «см. табл. 1.2.».



Список использованных источников должен содержать перечень источников, использованных при выполнении курсовой работы. Источники следует располагать в порядке появления ссылок в тексте записки (в порядке цитирования) или в алфавитном порядке. Сведения об источниках, включенных в список, необходимо оформлять в соответствии с требованиями ГОСТ 7.1-2003.

Приложения располагают в порядке появления ссылок на них в тексте. Каждое приложение следует начинать с новой страницы с указанием в правом верхнем углу слова «ПРИЛОЖЕНИЕ», написанного прописными буквами; оно должно иметь содержательный заголовок. Если в записке несколько приложений, их нумеруют арабскими цифрами, например, «ПРИЛОЖЕНИЕ 2». Рисунки, таблицы, формулы, помещенные в приложения, нумеруют арабскими цифрами в пределах каждого приложения, например, «Рис. П 1.2» – рисунок 2 приложения 1.

## **4. БИБЛИОГРАФИЧЕСКИЙ СПИСОК**

1. ГОСТ 7.32-2017 Отчет о научно-исследовательской работе. Общие требования и правила оформления М: Стандартинформ, 2018. 33 с.
2. ГОСТ Р 7.0.5-2008 Библиографическая ссылка. Общие требования и правила составления М: Стандартинформ, 2008. 23 с.
3. Шилдт Г. С++: базовый курс, 3-е издание. : Пер. с англ. – М.: «Издательский дом «Вильямс», 2005. – 624 с.
4. Вирт Н. Алгоритмы + структуры данных = программы. М.; Мир, 1985. – 281 с.
5. Шлее, М. Профессиональное программирование на С++ / М.Шлее .— СПб. : БХВ-Петербург, 2005 .— 544с. : ил. + 1 CD .— (В подлиннике).
6. Страуструп, Б. Язык программирования Си++: Спец. изд./ Б.Страуструп; Пер.сангл. С.Анисимова, М.Кононова; Подред. Ф.Андреева, А.Ушаков . — М. : Бином, 2004 .— 1098с.
7. Шилдт, Г. С# :Учеб.курс / Г.Шилдт; Пер.сангл.А.Падалки .— М. и др. : Питер, 2003 .— 512 с.
8. Секунов, Н.Ю. Разработка приложений на С++ и С# / Н.Ю.Секунов.— М. и др. : Питер, 2003 .— 608с. — (Б-ка программиста) .
9. Рихтер, Д.Windows для профессионалов : Создание эффективных Win32-приложений с учетом специфики 64-разрядной версии Windows / Д.Рихтер;Пер.сангл.:Ю.Е.Е.Купцевич,А.Р.Врублевский;Подобщ.ред.Ю.Е.Купцевича .— 4-е изд. — М.и др. : Питер, 2004 .— 722с.

## **ПРИЛОЖЕНИЕ**

В приложении представлены образец оформления основной части работы, формы титульного листа пояснительной записки курсовой работы, бланка задания и бланка рецензии.

## **Пример состава и оформления некоторых разделов основной части курсовой работы**

### **1 Постановка задачи**

Требуется создать библиотеку классов для работы с геометрическими объектами и составить задачи по вычислительной математике. Для этого необходимо:

- Изучить основы объектно-ориентированного программирования на языке C++.
- Рассмотреть основные понятия, такие как инкапсуляция, наследование, полиморфизм, класс, объект и другие.
- Изучить общий формат объявления класса, написание функций и перегрузку операторов.
- Изучить ограничения разных типов данных в языке C++.
- Изучить темы связанные с вычислительной математикой.

### **2 Теоретическая часть (фрагмент)**

Объектно-ориентированное программирование построено на понятии класса.

Класс — это базовая единица инкапсуляции, которая обеспечивает механизм создания объектов. Класс — это структурированный тип, включающий в себя в качестве элементов типизированные данные и функции, применяемые по отношению к этим данным.

Объекты — это экземпляры класса.

Инкапсуляция — это такой механизм программирования, который связывает воедино код и данные, которые он обрабатывает, чтобы обезопасить их как от внешнего вмешательства, так и от неправильного использования.

Полиморфизм (от греческого слова *polymorphism*, означающего "много форм") — это свойство, позволяющее использовать один интерфейс для целого класса действий.

...

*Раздел в данном примере сокращен. В каждом конкретном варианте объем этого раздела варьируется в зависимости от индивидуальной темы работы*

### 3 Создание библиотеки классов для работы с геометрическими объектами

Повторив основные понятия ООП, можем приступить к созданию классов для работы с геометрическими объектами.

Класс точка (вектор) будет содержать координаты  $x$  и  $y$  точки (вектора). Мы можем так же определить сложение и вычитание векторов, косое и скалярное произведение, квадрат расстояния между точками и другие функции и операторы.

Класс прямая создадим для хранения уравнения прямой вида:

$$a * x + b * y + c = 0.$$

Будем хранить коэффициенты  $a$ ,  $b$  и  $c$ . Так же можем получить эти коэффициенты по двум точкам, принадлежащим данной прямой.

Описание разработанных классов (*фрагмент*):

```
class pt
{
    double x, y;
    //long long int x, y;
public:
    pt(double x = 0, double y = 0) : x(x), y(y) {}
    //pt(long long int x = 0, long long int y = 0) : x(x), y(y) {}
    pt operator +(pt b)
    {
        pt res;
        res.x = x + b.x;
        res.y = y + b.y;
        return res;
    }
    pt operator -(pt b)
    {
        pt res;
        res.x = x - b.x;
        res.y = y - b.y;
        return res;
    }
    // и т.д.
};
```

#### 4 Составление и решение задач по вычислительной математике с использованием разработанных классов

Полученную библиотеку классов будем использовать в написании и решении задач по вычислительной математике.

##### *Пример оформления задачи*

##### **Задача XX.** Загадка планеты Титон

Планета Титон очень похожа на Землю. На ней тоже есть океаны, материки, острова. Планету населяют титонцы, которые больше всего на свете любят путешествовать. Но с недавних пор корабли титонцев стали пропадать в таинственных треугольниках, заполненных густым туманом. К счастью, загадка треугольников была разгадана. Чтобы безопасно проплыть сквозь такой треугольник, нужно плыть из любого угла точно по биссектрисе.

Треугольники появляются неожиданно, и корабли путешественников всегда оказываются на одном из его углов. В таком случае нужно незамедлительно менять курс. Помогите титонцам научиться безопасно преодолевать все таинственные треугольники.

По данным координат трех точек треугольника ABC определите коэффициенты  $a$ ,  $b$  и  $c$  для задания уравнения биссектрисы угла BAC (уравнение имеет вид  $ax + by + c = 0$ ).

Формат ввода. Три строки входных данных содержат по два вещественных числа, разделенных пробелом:  $x_i$  и  $y_i$  – координаты каждой точки треугольника ABC (все числа заданы в виде десятичного числа с плавающей точкой не более чем из 16 цифр, не превышают по модулю 10000).

Первая строка содержит координаты корабля попавших в беду путешественников (координату точки  $A(x_1; y_1)$  таинственного треугольника). Вторая и третья строки содержат координаты двух оставшихся точек треугольника –  $B(x_2; y_2)$  и  $C(x_3; y_3)$ .

Формат вывода. Выведите три целых числа  $a$ ,  $b$  и  $c$ , подходящих под уравнение биссектрисы ( $ax + by + c = 0$ ).

*Алгоритм решения: ....*

Рассмотрим *группы тестов* для проверки решения данной задачи. Коэффициент  $c$  биссектрисы равен 0. Биссектриса проходит через начало координат.

0 0	0 0	0 0
0 1	4 6	0 3.5

-1 0	4 -1	2.5 0
-0.7071067932881648	-0.3605966625409265	-0.7071067811865475
-0.7071067932881648	0.9327218211049204	0.7071067811865475
0.0000000000000000	0.0000000000000000	0.0000000000000000

Коэффициент  $b$  равен 0. Биссектриса параллельна оси Ох. При проверке ответов с помощью чекера этот случай стоит рассматривать отдельно, чтобы избежать ошибок, связанных с делением на 0.

0 0	0 0	2.2 0.1
-3 3	3 -3	1.8 0.7
10 10	-6 -6	2.6 0.7
-1.0000000000000000	1.0000000000000000	-1.0000000000000000
-0.0000000000000001	0.0000000000000000	-0.0000000000000001
0.0000000000000000	-0.0000000000000000	2.2000000000000002

Коэффициент  $a$  равен 0. Биссектриса параллельна оси Оу. Случай так же следует рассматривать отдельно.

0 0	0 0	2.2 0.1
4 4	-3 3	2.4 -0.2
2 -2	-8 -8	2.6 0.7
0.0000000000000000	-0.0000000000000001	0.0000000000000003
1.0000000000000000	-1.0000000000000000	1.0000000000000000
0.0000000000000000	0.0000000000000000	-0.1000000000000007

Все коэффициенты не равны 0. Биссектриса не параллельна осям координат.

-3 1	0.65 0.9	2.4 -3.6
-10 -3	4.02 7.04	5.8 -4.8
-6 -7	8.66 2.9	2.8 -4.2
0.7614751617463874	-0.6105515760049598	0.6139406135149205
-0.6481940897935687	0.7919764977812533	0.7893522173763262
2.9326195750327311	-0.3159203235999042	1.3682105101189654

Так как коэффициенты могут отличаться вплоть до умножения на какую-либо константу, для проверки правильности ответов следует использовать чекер:

```
#include "testlib.h"
#include <cmath>

using namespace std;
const double eps = 1e-8;

int main(int argc, char* argv[]) {
    registerTestlibCmd(argc, argv);
```

```

double a = ouf.readDouble();
double b = ouf.readDouble();
double c = ouf.readDouble();
double A = ans.readDouble();
double B = ans.readDouble();
double C = ans.readDouble();
double x1 = 0, x2 = 10;
double y1, y2;
if (abs(B) > eps) //Коэффициент b не равен 0
{
    y1 = (-A * x1 - C) / B;
    y2 = (-A * x2 - C) / B;
    if ((fabs(a * x1 + b * y1 + c) < eps) && (fabs(a * x2 + b * y2 + c) < eps)) {
        quitf(_ok, "The result is correct.");
    }
    else {
        quitf(_wa, "Wrong answer: expected = %f, %f, %f, found = %f, %f, %f", A,
B, C, a, b, c);
    }
}
else {
    if (abs(A) > eps) //Коэффициент a не равен 0
    { x1 = x2 = -C / A;
        if ((fabs(a * x1 + c) < eps) && (fabs(b) < eps)) {
            quitf(_ok, "The result is correct.");
        }
        else {
            quitf(_wa, "Wrong answer: expected = %f, %f, %f, found = %f, %f, %f", A, B
, C, a, b, c);
        }
    }
}
}
}

```

Решение с использованием класса pt.

```

int main()
{
    pt a, b, c, p;
    cin >> a >> b >> c;
    pt v1 = b - a, v2 = c - a;
    v1 = v1 / (sqrt(v1.dist2()));
    v2 = v2 / (sqrt(v2.dist2()));
    p = v1 + v2;
    p = p / sqrt(p.dist2());
    pt a0;
    line A(a0, p);
    cout << fixed << setprecision(16) << A << ' ' << cross(a, p);
    return 0;
}

```

*Пример описания групп тестов:*

Тесты:

№1 - 5 – а, b, c, l, r в пределах  $\pm 100$ , запросов не более 10, переменная а от 0 до 1 с точностью  $10^{-1}$ .

№6 - 10 – а, b, c, l, r в пределах  $\pm 100$ , запросов не более 100, переменная а от 0 до 1 с точностью  $10^{-2}$ .

№11 - 15 – а, b, c в пределах  $\pm 1000$ , l, r в пределах  $\pm 100$ , запросов не более 1000, переменная а от 0 до 1 с точностью  $10^{-3}$ .

№16 - 30 – а, b, c в пределах  $\pm 1000$ , l, r в пределах  $\pm 10000$ , запросов не более 100000, переменная а = 0.

**МИНОБРНАУКИ РОССИИ**  
**Федеральное государственное бюджетное**  
**образовательное учреждение высшего образования**  
**«Тульский государственный университет»**

Кафедра информационной безопасности

**ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

к курсовой работе по дисциплине

**«Языки программирования»**

на тему

**«Префиксные суммы»**

Автор работы \_\_\_\_\_ студент гр. 230711  
(дата, подпись)

Павлова В.С.  
(фамилия и инициалы)

Руководитель работы \_\_\_\_\_  
(дата, подпись) \_\_\_\_\_ (должность) \_\_\_\_\_ (фамилия и инициалы)

Работа защищена \_\_\_\_\_ с оценкой \_\_\_\_\_  
(дата)

Члены комиссии \_\_\_\_\_  
(дата, подпись) \_\_\_\_\_ (должность) \_\_\_\_\_ (фамилия и инициалы)

\_\_\_\_\_  
(дата, подпись) \_\_\_\_\_ (должность) \_\_\_\_\_ (фамилия и инициалы)

\_\_\_\_\_  
(дата, подпись) \_\_\_\_\_ (должность) \_\_\_\_\_ (фамилия и инициалы)

Тула 2022



УТВЕРЖДАЮ  
Зав. кафедрой ИБ

\_\_\_\_\_ А.А. Сычугов  
« \_\_\_\_ » \_\_\_\_\_ 20\_\_ г.

**ЗАДАНИЕ**  
на курсовую работу по дисциплине  
**«Языки программирования»**

студенту гр. \_\_\_\_\_  
(фамилия, имя, отчество)

Тема работы

---

---

---

Входные данные

---

---

---

---

---

Задание получил \_\_\_\_\_  
(подпись) (дата)

График выполнения работы

---

---

---

---

---

---

---

Замечания консультанта

---

---

---

К защите. Консультант работы \_\_\_\_\_  
(подпись) (дата)

## РЕЦЕНЗИЯ

на курсовую работу (курсовой проект) студента Тульского государственного университета,

гр. \_\_\_\_\_  
(фамилия, имя, отчество)

по дисциплине \_\_\_\_\_

на тему: \_\_\_\_\_

Тематика работы (задание) \_\_\_\_\_ профилю (направленности)  
(соответствует/не соответствует)  
обязательной программы, а также в полной мере способствует формированию необходимых компетенций (установленных в рабочей программе) у обучающихся.

Содержание работы \_\_\_\_\_ заданной тематике.  
(соответствует/не соответствует)

Объем работы \_\_\_\_\_ для раскрытия темы.  
(соответствует/не соответствует)

Оформление работы \_\_\_\_\_ установленным требованиям.  
(соответствует/не соответствует)

Тема работы раскрыта в \_\_\_\_\_ мере.  
(полной/неполной)

Использованная при выполнении работы (проекта) литература \_\_\_\_\_  
(актуальна/не актуальна)

Качество приложений (при наличии) \_\_\_\_\_ установленным требованиям.  
(соответствует/не соответствует)

Замечания: \_\_\_\_\_  
(отсутствуют или перечисляются замечания рецензента к работе)

Качество выполнения работы свидетельствует о \_\_\_\_\_ уровне  
(недостаточном/пороговом/повышенном)  
сформированности необходимых компетенций (установленных в рабочей программе).

Работа выполнена на \_\_\_\_\_ уровне и, при соответствующей защите  
(высоком/среднем/низком)  
может быть оценена на << \_\_\_\_\_ >>.

Рецензент \_\_\_\_\_ / \_\_\_\_\_ /  
подпись должность, звание, Ф.И.О.

Дата \_\_\_\_\_