|  |  |  |
| --- | --- | --- |
| Министерство науки и высшего образования РФ  Федеральное государственное автономное образовательное учреждение высшего образования  «Пермский государственный национальный исследовательский университет» | | |
|  | Институт компьютерных наук и технологий | |
| **ОТЧЁТ**  по лабораторной работе №4  по дисциплине «Языки программирования»  Вариант 5 | | |
|  | | Работу выполнил  студент группы ПМИ-9,10-2024 2 курса  Казаков Н.С.  «14» Ноября 2024 г. |
| Работу проверила  Ракина В.Д. преподавателя  «\_\_\_» \_\_\_\_\_\_\_\_\_ 2024 г. |
| Пермь 2024 | | |

СОДЕРЖАНИЕ

[Задание 1 4](#_Toc182828699)

[Текст задания 4](#_Toc182828700)

[Тестирование 5](#_Toc182828701)

[Код программы 5](#_Toc182828703)

[Задание 2 6](#_Toc182828704)

[Текст задания 6](#_Toc182828705)

[Алгоритм решения 6](#_Toc182828706)

[Демонстрация возможностей класса 6](#_Toc182828707)

[Тестирование 7](#_Toc182828708)

[Код программы 7](#_Toc182828710)

[Класс LineSegment 8](#_Toc182828711)

# Задание 1

## Текст задания

Название класса:

LineSegmentПоля:double x double y (координаты начала и конца отрезка на координатной прямой) Методы:

Определить, попадает ли заданное число в отрезок. Результат должен быть типа bool. Алгоритм решения

1. Ввод данных:

* Пользователь вводит начало (start) и конец (end) отрезка.
* Пользователь вводит число (testPoint) для проверки.

1. Создание объекта:

* Создается объект класса LineSegment с переданными началом и концом отрезка.

1. Проверка попадания:

* Используется метод isWithinSegment класса LineSegment. Он проверяет, находится ли переданное число testPoint в пределах отрезка [x, y] или [y, x].

1. Вывод результата:

* Выводится сообщение о том, попадает ли число в отрезок.

## Тестирование

## Изображение выглядит как текст, Шрифт, снимок экрана Автоматически созданное описаниеИзображение выглядит как текст, Шрифт, снимок экрана, черный Автоматически созданное описаниеИзображение выглядит как текст, Шрифт, снимок экрана Автоматически созданное описание

## Код программы

[lab-4/code.cpp at main · KNs4/lab-4](https://github.com/KNs4/lab-4/blob/main/code.cpp)

# Задание 2

## Текст задания

Название класса  
LineSegment

Методы

Унарные операции: ! - вычислить длину отрезка, результат должен быть типа double; ++ увеличить координаты границ отрезка на 1. Операции приведения типа: int (явная) – результатом является целая часть координаты х; double (неявная) – результатом является координата y. Бинарные операции: + int d – координаты границ увеличиваются на число d; > int d – результат равен true, если целое число попадает в заданный отрезок и false–в противном случае.

## Алгоритм решения

## Демонстрация возможностей класса

1.Унарные операции:

* Оператор ! вычисляет длину отрезка как модуль разности между x и y.
* Инкремент ++ увеличивает границы отрезка на 1.

2.Приведение типов:

* Приведение объекта к int возвращает начальную точку (x).
* Приведение к double возвращает конечную точку (y).

3.Бинарные операции:

* Сдвиг отрезка на число (+) создает новый объект, увеличивая обе границы отрезка на указанное значение.
* Оператор > проверяет, находится ли заданное число внутри отрезка.

4.Вывод результатов:

* Показаны результаты работы методов и перегруженных операторов, подтверждающие функциональность класса.

## Тестирование

## Изображение выглядит как текст, снимок экрана, Шрифт Автоматически созданное описаниеИзображение выглядит как текст, снимок экрана, Шрифт Автоматически созданное описаниеИзображение выглядит как текст, снимок экрана, Шрифт Автоматически созданное описание

## Код программы

[lab-4/code.cpp at main · KNs4/lab-4](https://github.com/KNs4/lab-4/blob/main/code.cpp)

# Класс LineSegment

1. **Поля:**

* **x и y:**
  + Поля класса, которые хранят координаты начала (x) и конца (y) отрезка.
  + Объявлены как private, то есть доступны только методам класса.

1. **Конструкторы:**
2. **По умолчанию:**
   * Инициализирует объект со значениями [0, 0].
3. **С параметрами:**
   * Позволяет задать начало и конец отрезка при создании объекта.
4. **Конструктор копирования:**
   * Создает новый объект, копируя значения полей x и y из другого объекта.
5. **Методы:**
6. **Геттеры и сеттеры:**
   * Предназначены для получения (getX, getY) и изменения (setX, setY) значений полей x и y.
7. **Логический метод:**
   * isWithinSegment проверяет, находится ли переданное число в пределах отрезка. Учитывает, что x может быть больше y.
8. **Перегруженные операторы:**

* **Унарные операторы:**
  + ! возвращает длину отрезка как модуль разницы между x и y.
  + ++ увеличивает обе границы (x и y) на единицу.
* **Приведение типов:**
  + Приведение к int возвращает начальную точку (x) как целое число.
  + Приведение к double возвращает конечную точку (y) как число с плавающей точкой.
* **Бинарные операторы:**
  + + создает новый отрезок, сдвинув границы на заданное значение.
  + > проверяет, находится ли число в пределах отрезка.
* **Оператор вывода (<<):**
  + Формирует строку с описанием объекта, например: LineSegment: [x, y]