Министерство образования Республики Беларусь

Учреждение образования

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет Информационных технологий и управления

Кафедра Интеллектуальных информационных технологий

ОТЧЁТ

Лабораторная работа №1

Выполнил: Промчук Д. В.

Проверил: Никифоров С. А.

Минск 2022

Тема: изучение основ объектно-ориентированного программирования на языке программирования С++.

Задание: реализовать на языке С++ один из нижеперечисленных вариантов и написать и сгенерировать документацию при помощи doxygen.

Вариант 1.7 Прямоугольник:

Описать класс прямоугольника со сторонами, параллельными осям координат. Вершины прямоугольников имеют должны иметь целочисленные координаты. Класс должен реализовывать следующие возможности:

∙ получение координат вершин;

∙ перемещение;

∙ изменение размера;

∙ увелечение размера на единицу по каждой из осей;

∙ уменьшение размера на единицу по каждой из осей;

∙ построение наименьшего прямоугольника, содержащего два заданных прямоугольника;

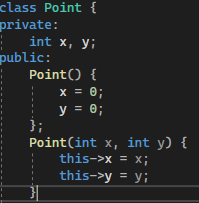
∙ построение наименьшего прямоугольника, содержащего два заданных прямоугольника, с присваиванием;

∙ построение прямоугольника, являющегося общей частью (пересечением) двух прямоугольников;

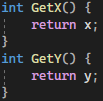
∙ построение прямоугольника, являющегося общей частью (пересечением) двух прямоугольников, с присванием;

Поскольку вся документация с использованием doxygen прилагается к работе, я уточню основные моменты в данном отчёте

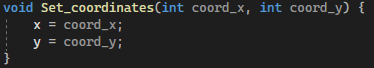
Файл Header.h:



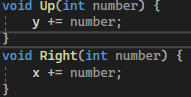
Создан класс для работы с точкой, которая и является вершиной прямоугольника. Точка хранит в себе 2 координаты: Х и Y, с которыми и происходит работа.



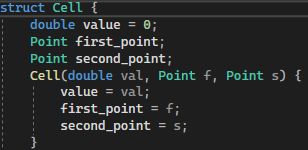
Первые 2 метода представляют из себя простейшие “геттеры”, возвращающие соответствующие координаты точки.



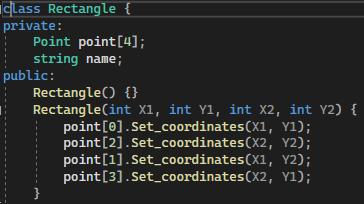
Данный метод представляет из себя “сеттер”, устанавливающий точке полученные координаты.



Данные методы изменяют значение соответствующей координаты на полученное значение, что при использовании отрицательных чисел также позволяет их уменьшить.

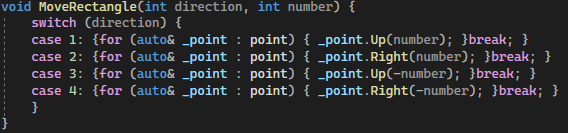


Данная структура создана для хранения адресов 2 точек и расстояния между ними.

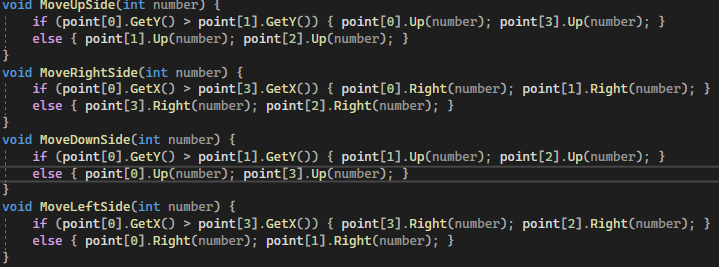


Основной класс в данной работе. Хранит в себе массив из 4 объектов класса Point, соответствующих 4 своим вершинам и имя прямоугольника. Конструктор по умолчанию принимает координаты 2 диагональных точек, что позволяет использовать меньше памяти и облегчает работу для пользователя, поскольку ему не надо считать и вводить координаты 2 точек, ведь они уже фактически есть в 2 других точках.

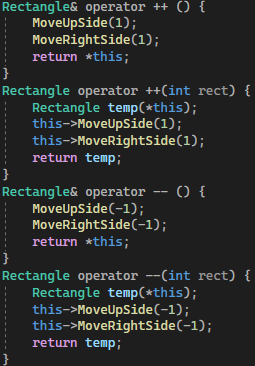
У класса реализованы “сеттеры” и “геттеры” по аналогии с таковыми класса Point.



Данный метод изменяет значения координат на number пунктов в соответствии с первым полученным значением: 1-вверх, 2-вправо, 3-вниз, 4-влево.



Каждый из данных методов передвигает какую-либо сторону на полученное значение единиц.



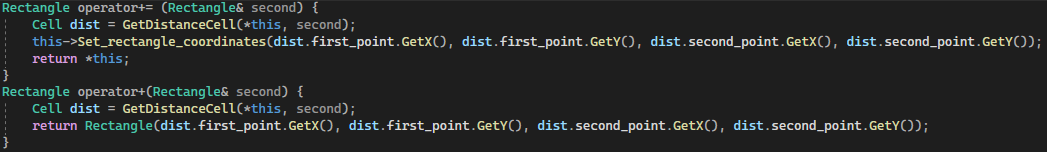
Данные перегрузки операторов увеличивают/уменьшают соответствующие значения X или Y у верхней и правой сторон прямоугольника на 1.

Предекремент и преинкремент сначала выполняют свою работу, а после возвращают значение.

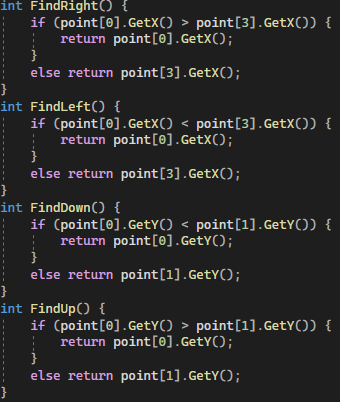
Постдекремент и постинкремент сначала возвращают значение, а после выполняют свою работу. Здесь это реализовано копирующей инициализацией прямоугольника в начале работы с созданием временного объекта, который по окончанию работы и возвращает метод.



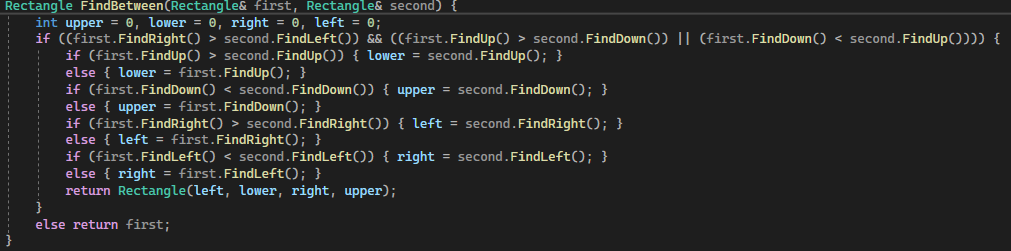
Данный метод позволяет получить адрес соответствующей точки в массиве вершин прямоугольника.



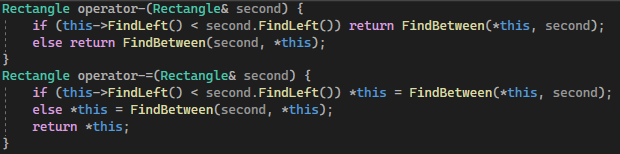
Экземпляр dist структуры cell хранит в себе адреса памяти саммых удалённых точек 2 прямоугольников, поскольку они и станут диагональными точками нового прямоугольника. Данные перегрузки возвращают прямоугольник, который содержит в себе оба заданных прямоугольника. Перегрузка += устанавливает новые координаты для основного прямоугольника, а перегрузка + создаёт новый прямоугольник с необходимыми координатами.



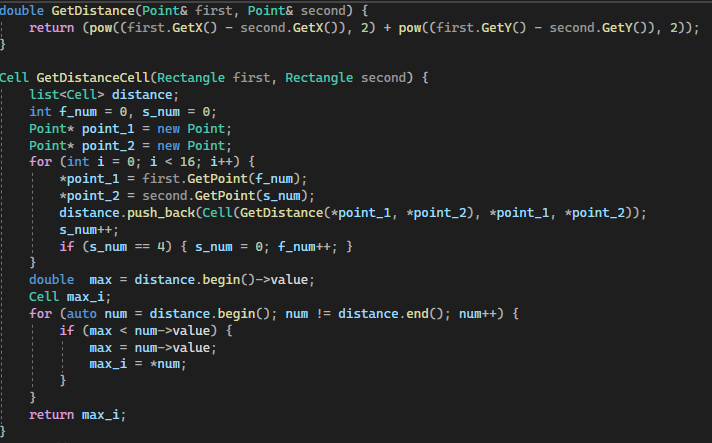
Данные методы находят в прямоугольнике координаты соответствующих сторон.



Данный метод проверяет 2 прямоугольника на наличие точек пересечения, а после создаёт новый, полученный из координат точек их пересечений.



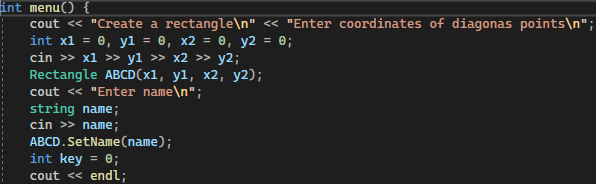
Данные операторы вызывают предыдущий метод, гарантируя что первый прямоугольник окажется левее относительно второго. Отличия между ними такие же, как и отличия + и +=



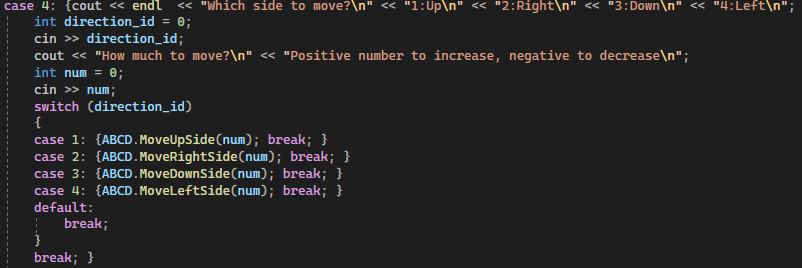
Функция GetDistanceCell получает 2 объекта класса Rectangle, создаёт список из экземпляров структуры Cell, вычисляя для каждой значения, а после находит экземпляр с самым большим расстоянием и возвращает данный экземпляр.

Файл Laba1\_ppois.cpp:

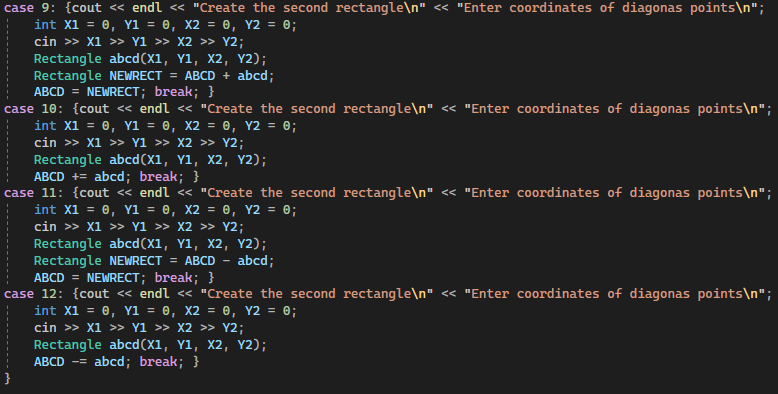
Данный файл содержит лишь меню, и функцию main вызывающую его.



В начале работы нужно ввести данные для создания экземпляра класса Rectangle и последующей работы с ним.

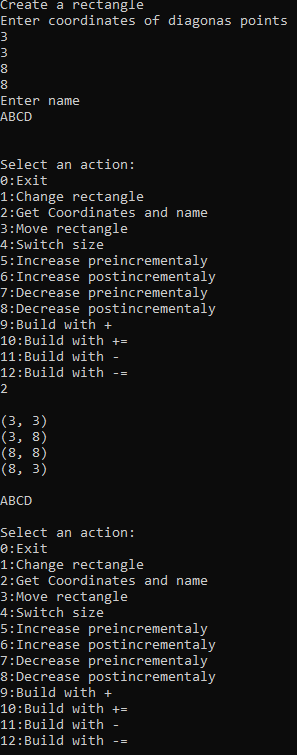


В данном месте программа выводит пользователю какая цифра за какое направление отвечает, а так же что уменьшение размера происходит путём ввода отрицательных чисел.



Поскольку в задании не сказано про хранение нескольких объектов класса Rectangle, в каждом случае из вышеперечисленных будет сначала создаваться прямоугольник, а по завершению координаты присвоены первоначальному прямоугольнику.

Стартовое меню в консоли:



Для наглядности создан прямоугольник и выведены его координаты.

**Вывод:** в ходе выполнения работы изучены основные понятия объектно-ориентированного програмирования на С++, а так же реализована программа, выполняющая некоторые преобразования с прямоугольниками.