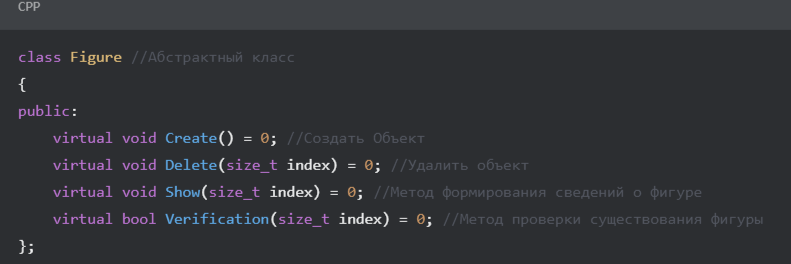
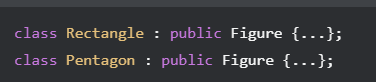
Основные компоненты программыАбстрактный класс Figure:

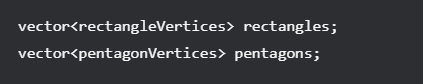


**Это абстрактный класс, который задает интерфейс для всех фигур. Методы объявлены как виртуальные, что позволяет производным классам реализовывать свои версии этих методов (полиморфизм)**

**Классы Rectangle и Pentagon:Оба класса наследуются от абстрактного класса Figure и реализуют его методы^**

****

**Каждый класс имеет вектор (контейнер) для хранения своих фигур:**

****

**Контейнеры**

**:std::vector используется для хранения созданных прямоугольников и пятиугольников. Это динамический массив, который позволяет добавлять и удалять элементы.Обе фигуры могут быть созданы до тех пор, пока не исчерпают доступную память, что означает, что их количество практически не ограничено (за исключением свободной памяти)**

**Методы в классе Rectangle**

**1Create:**

****

**Запрашивает координаты двух вершин прямоугольника и добавляет его в вектор.**

**2Delete**



Удаляет прямоугольник по индексу из вектора, проверяя, не превышает ли индекс допустимые значения.

3Show:



Выводит информацию о прямоугольнике на экран.

4Verification:



Проверяет, существует ли фигур по данному индексу.

5setAllVertices и GetVertex

методы:setAllVertices перемещает вершины прямоугольника на заданные значения x и y, что позволяет изменять позицию уже существующих фигур.Каждый из этих методов реализует конкретное поведение и логику, специфичную для прямоугольника, и содержит проверки для предотвращения получения некорректных индексов.

Методы в классе PentagonАналогично, методы в классе Pentagon делают то же самое, но для пятиугольников. Они работают с пятью вершинами:Create, Delete, Show, Verification, и методы для получения вершин

Главное тело программы.

В главном цикле программы пользователь может выбирать, какую фигуру создать, удалить, переместить и даже проверить пересечение двух фигур с помощью функции Intersect.

1Установка кодировки:



Эта команда устанавливает кодировку консоли для корректного отображения символов кириллицы. Она позволяет работать с текстом на русском языке, выводимым в консоль.

2Создание объектов:



Здесь создаются экземпляры классов Pentagon и Rectangle, которые будут использоваться для управления соответствующими фигурами.

3Переменная choice:



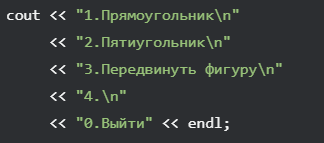
Эта переменная используется для хранения выбора пользователя в меню. Изначально она установлена в -1, чтобы войти в цикл while.

4Основной цикл программы:



Цикл выполняется, пока пользователь не введет 0, что позволяет программе продолжать работу, предоставляя пользователю возможности взаимодействовать с фигурами.

Внутри этого цикла:Переменная choice переопределяется в -1, чтобы снова запрашивать выбор пользователя.Публикуется меню для выбора действий



Пользователь выбирает действие, и выполнение программы переходит к проверке выбора.Проверка выбора с помощью switch:

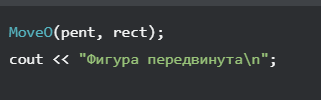


Конструкция switch управляет выполнением различных блоков кода в зависимости от введенного пользователем числа:

Случай 1: Создание и управление прямоугольниками:В этом блоке пользователь может создать новый прямоугольник, удалить его, вывести информацию или проверить, существует ли фигура.

Случай 2: Аналогично для пятиугольников:Позволяет делать все те же операции, но для объекта pent.

Случай 3: Перемещение фигуры:



Функция MoveO вызывается с объектами pent (пятиугольник) и rect (прямоугольник) в качестве аргументов. Внутри этой функции пользователь выбирает, какую фигуру хочет переместить, и на сколько.

Случай 4: Проверка пересечения:Этот случай отсутствует в коде, но если вы планируете добавить его, он может включать вызов функции для пересечения двух фигур (например, в случае с функцией Intersect).

default:Обрабатывает случаи, когда пользователь ввел некорректный вариант, не предусмотренный в меню.

Переменные и их роли

choice: Для хранения выбора пользователя и управления различными задачами.

pent и rect: Экземпляры классов Pentagon и Rectangle, которые содержат операции, специфичные для работы с пятьюугольниками и прямоугольниками.

Заключение

Главное тело программы управляет пользовательским интерфейсом, взаимодействуя с объектами Pentagon и Rectangle. Оно предлагает пользователю различные возможности создания, удаления и управления фигурами, а также позволяет перемещать их и проверять существование. Ключевыми аспектами являются управление выборами и использование полиморфизма и инкапсуляции через определенные методы, которые обеспечивают правильную работу с фигурами.

Полиморфизм

Полиморфизм проявляется в том, что мы можем работать с объектами классов Rectangle и Pentagon через интерфейс абстрактного класса Figure. Это позволяет программировать гибко и добавлять новые фигуры, реализуя лишь интерфейс Figure.

Заключение

Таким образом, программа соответствует принципам ООП, обеспечивая гибкость, легкость в расширении (добавление новых фигур возможно), и защищая от ошибок за счет использования контейнеров и проверки индексов. По умолчанию программа не содержит механизмов, ограничивающих количество создаваемых фигур, кроме памяти, поэтому можно бесконечно создавать фигуры до исчерпания памяти

1УОсновной цикл программы:

Эта переменная используется для хранения выбора пользователя в меню. Изначально она