

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ
ФЕДЕРАЦИИ МОСКОВСКИЙ АВИАЦИОННЫЙ ИНСТИТУТ
(НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)

ЛАБОРАТОРНАЯ РАБОТА №2
по курсу объектно-ориентированное программирование I семестр, 2021/22
уч. год

Студент *Шандрюк Пётр Николаевич, группа М8О-208Б-20*

Преподаватель *Дорохов Евгений Павлович*

Условие

Задание: Вариант 27: Прямоугольник, очередь. Необходимо спроектировать и запрограммировать на языке C++ класс-контейнер первого уровня, содержащий одну фигуру, согласно вариантам задания. Классы должны удовлетворять следующим правилам:

1. Требования к классу фигуры аналогичны требованиям из лабораторной работы 1.
2. Классы фигур должны содержать набор следующих методов:
 - Перегруженный оператор ввода координат вершин фигуры из потока `std::istream` («»). Он должен заменить конструктор, принимающий координаты вершин из стандартного потока.
 - Перегруженный оператор вывода в поток `std::ostream` («»), заменяющий метод `Print` из лабораторной работы 1.
 - Оператор копирования (`=`)
 - Оператор сравнения с такими же фигурами (`==`)
3. Класс-контейнер должен содержать объекты фигур “по значению” (не по ссылке).
4. Класс-контейнер должен содержать набор следующих методов:
 - `push` - вставка в очередь
 - `pop` - удаление из очереди
 - `top` - первый элемент в очереди
 - `empty` - пустая ли очередь
 - `size` - количество элементов в очереди
 - Перегруженный оператор вывода в поток `std::ostream` («»)

Описание программы

Исходный код лежит в 9 файлах:

1. `main.cpp`: основная программа, взаимодействие с пользователем посредством команд из меню
2. `tqueueitem.h`: описание класса предмета очереди
3. `point.h`: описание класса точки
4. `tqueue.h`: описание класса очереди

5. rectangle.h: описание класса треугольника, наследующегося от figures
6. point.cpp: реализация класса точки
7. tqueue.cpp: реализация класса очереди
8. rectangle.cpp: реализация класса треугольника
9. tqueueitem.cpp: реализация класса предмета очереди

Дневник отладки

Недочёты

Выводы

Я научился создавать простые динамические структуры данных, а также работать с объектами, передаваемыми "по значению".

Исходный код

tqueue.h

tqueue.cpp

tqueueitem.h

tqueueitem.cpp

point.h

point.cpp

trinagle.h

triangle.cpp

main.cpp