**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В.И. Ульянова (Ленина)**

**Кафедра Вычислительной техники**

отчет

**по лабораторной работе №1**

**по дисциплине «Алгоритмы и структуры данных»**

Тема: «Работа с иерархией объектов: наследование и полиморфизм»

|  |  |  |
| --- | --- | --- |
| Студенты гр. 1305 |  | Данилов А.С.  Серкин Д.А. |
| Преподаватель |  | Колинько П.Г. |

Санкт-Петербург

2023

**Цель работы**

Завершение знакомства с возможностями объектного программирования.

**Задание (вариант 2)**

Доработать модуль shape.cpp, добавив в коллекцию параллелограмм с крестом. Расположить его на позиции 1 — галстук, 14 — шишак.

**Описание получившейся иерархии классов**

parallelogram\_x

reflectable

rectangle

rotatable

shape

**Добавленные классы**

Был добавлен класс, описывающий фигуру «Параллелограмм с крестом»:

|  |  |  |  |
| --- | --- | --- | --- |
| Наименование | Вид | Отражение | Поворот |

|  |  |  |  |
| --- | --- | --- | --- |
| Параллелограмм с крестом |  | Да | Да |

Его название - parallelogram\_x. Он наследуется от классов rectangle и reflectable. Крайние точки из класса rectangle позволяют составить параллелограмм и вписать в него крест.

**Переопределённые функции-члены**

Были определены 4 угловые точки (corner1…4), координаты которых рассчитываются с помощью основных точек sw и ne из прямоугольника. При повороте фигуры координаты этих точек нужно пересчитывать, поэтому были добавлены rot\_corner1…4.

Чтобы понять, какие точки использовать для отрисовки, были определены функции l\_rotate() и r\_rotate(),которые меняют числовую переменную, а также вызывают функции для поворота из класса rectangle. Переменные учитывает переопределённая функция draw().

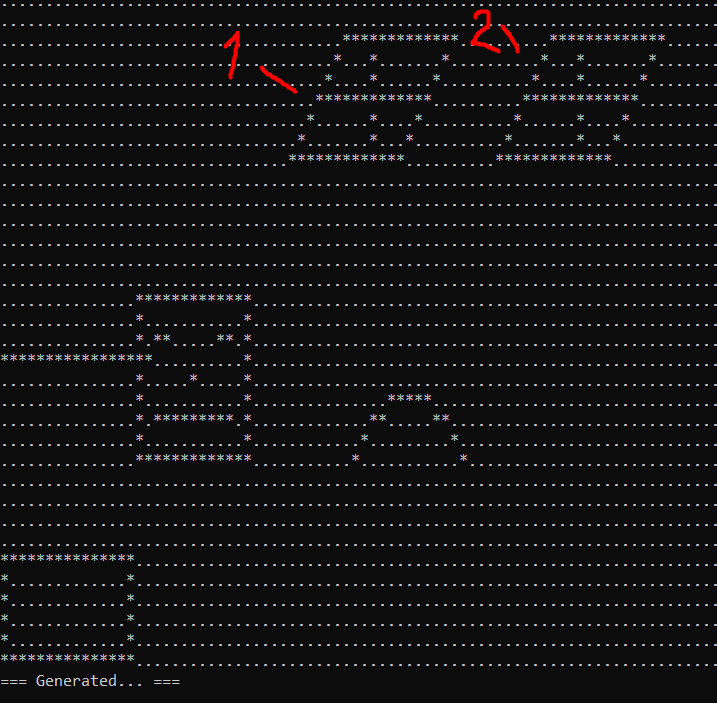
Для отражения параллелограмма были определены функции flip\_vertically() и flip\_horisontally().

**Недоступные функции-члены**

С помощью delete были сделаны недоступными специальные функции-члены, генерируемые компилятором.

**Контрольный пример**

1. Начальный набор фигур.



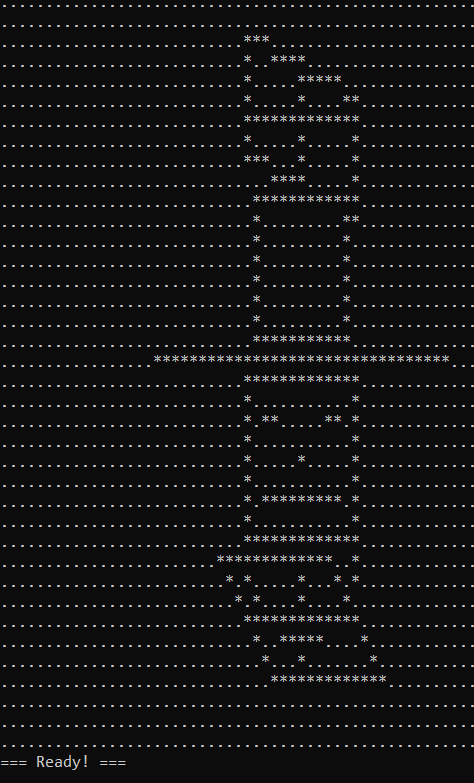
1. Фигура 1 отражена по вертикали.



1. Фигура 2 повёрнута вправо.



1. Результат

****

**Выводы**

В процессе выполнения данного задания мы изучили принципы полиморфизма и наследования классов. Благодаря их использованию можно, например, создавать более общие классы, основанные на уже существующих, и добавлять в них новые функции.

**Список использованных источников**

1. П.Г. Колинько – «Пользовательские контейнеры» учебно-метод. пособие, 2023 г.

**Доработанный текст shape.cpp**

В файле parallelogram\_x.h.