

Программирование на языке Python
Лабораторная работа № 3. Объектно-ориентированное программирование,
исключения и их обработка.

Каждое задание должно быть загружено на личный git-репозиторий отдельным коммитом, возможно, не одним. Все коммиты должны иметь осмысленные названия и описания того, что в них выполнено. Защита работы возможна на любом лабораторном занятии. Наличие выполненных работ учитывается на зачете.

Если две и более задачи выполнены в один коммит, работа не проверяется. Если все коммиты сделаны в один час, работа не проверяется.

Общее задание. Познакомиться с объектно-ориентированным программированием и исключениями и их обработкой. Для этого необходимо выполнить следующие задания.

Задание 1. Написать программу, демонстрирующую работу с объектами двух типов T1 и T2, для чего создать систему соответствующих классов. Каждый объект должен иметь идентификатор (в виде произвольной строки символов) и одно или несколько полей для хранения состояния (текущего значения) объекта.

Перечень типов объектов: Треугольник (Triangle), Квадрат (Quad), Прямоугольник (Rectangle), Четырехугольник (Tetragon), Пятиугольник (Pentagon).

Перечень дополнительных методов:

- move() переместить объект на плоскости.
- compare(T1 t1, T2 t2) сравнение объектов по площади.
- is_intersect(T1 t1, T2 t2) определяет факт пересечения объектов.
- is_include(T1 t1, T2 t2) определяет факт включения t2 в t1.

Вариант: $((N - 1) \% 15) + 1$, где N – номер по списку.

Вариант	T1	T2	Методы
1	Triangle	Quad	move, compare
2	Quad	Pentagon	move, is_intersect
3	Triangle	Rectangle	move, compare
4	Triangle	Rectangle	move, is_intersect
5	Rectangle	Pentagon	move, is_include
6	Triangle	Tetragon	move, compare
7	Triangle	Tetragon	move, is_intersect
8	Triangle	Tetragon	move, is_include
9	Triangle	Pentagon	move, compare
10	Triangle	Pentagon	move, is_intersect
11	Rectangle	Quad	is_intersect, compare
12	Pentagon	Triangle	compare, is_intersect

13	Rectangle	Pentagon	is_include, compare
14	Quad	Pentagon	compare, is_intersect
15	Rectangle	Pentagon	is_intersect, is_include

Задание 2. Необходимо предусмотреть генерацию и обработку исключений для возможных ошибочных ситуаций.