

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ
ФЕДЕРАЦИИ

ПРЕЗИДЕНТСКИЙ ФИЗИКО-МАТЕМАТИЧЕСКИЙ ЛИЦЕЙ №
239

ОТЧЁТ ПО ГОДОВОМУ ПРОЕКТУ

Ученик:

Попельшко Аким

Преподаватель:

Клюнин Алексей Олегович

Класс:

10-3

Санкт-Петербург
2017

Содержание

1	Постановка задачи	3
2	Алгоритм решения задачи	3
2.1	Базовые структуры данных	3
2.2	Построение алгоритма	3

1 Постановка задачи

Заданы два множества точек на плоскости. Построить пересечения и разность этих множеств.

Используемые программы: GitHub, IntelliJ IDEA, TeXstudio, OpenGL.

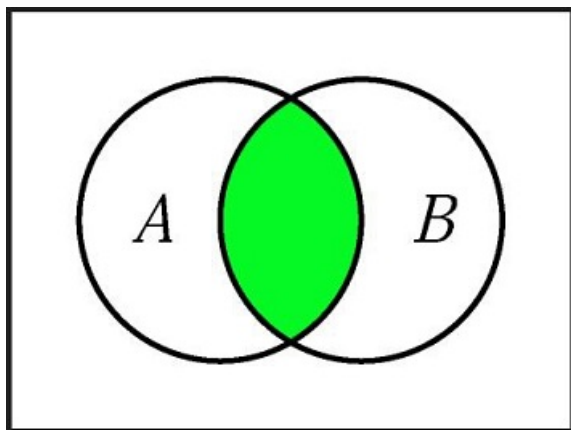


Рис. 1: Пересечение двух множеств

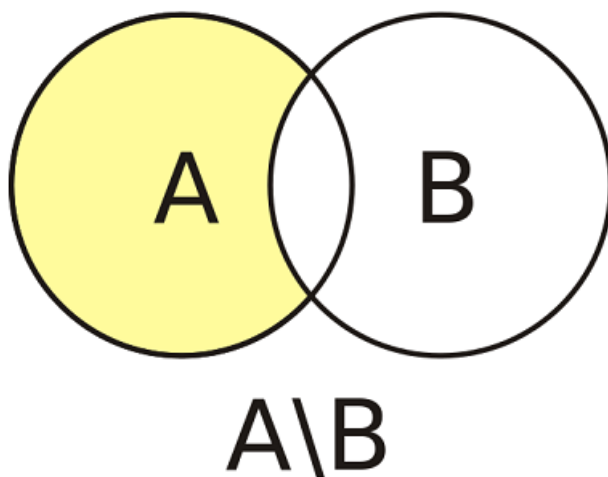


Рис. 2: Разность двух множеств

2 Алгоритм решения задачи

2.1 Базовые структуры данных

Класс Dot (точка) с двумя полями: x и y типа `real`, обозначающих координаты точки.

Класс Set (множество) с двумя полями: число `cardinality` типа `integer`, которое определяет количество элементов множества, и массив `arr` из элементов класса `Dot`. Длина массива равна `cardinality`.

2.2 Построение алгоритма

На вход подаётся целое число (m), определяющее количество точек в множестве A , затем несколько пар вещественных чисел, являющихся координатами точек множества A . Потом снова подаётся целое число (n), определяющее количество точек в множестве B и ещё несколько пар вещественных чисел, являющихся координатами точек множества B .

В самой программе сначала создаётся объект `a` класса `Set` с заданным числом элементов с помощью конструктора от поля `cardinality`. В цикле `for` каждая пара чисел считывается, затем с помощью конструктора от `x` и `y` создаётся объект класса `Dot`, и этот объект сразу же записывается в массив `a.arr`. Когда количество точек достигает `m`, цикл `for` заканчивается, и затем точно такие же действия выполняются для множества `B`.

Когда оба множества записаны, создаются два объекта класса `Set`: `intersection` с длиной массива, равной меньшей из чисел `a.cardinality` и `b.cardinality`; и `complement` с длиной массива, равной `a.cardinality`. Затем программа приступает к заполнению этих множеств.

К помощью цикла `foreach` программа ищет каждый элемент массива `a.arr` в массиве `b.arr` (с помощью метода `equals`), и, если эта точка есть в массиве `b.arr`, записывает её в массив `intersection.arr`, а если этой точки там нет – в массив `complement.arr`. Затем оба множества (`intersection` и `complement`) выводятся на экран.