**Тестовое задание по теме «Объектно-ориентированное программирование на языке Java с применением циклических алгоритмов обработки массивов объектов»**

**Указания к выполнению задания:**

* Всеклассыдолжныудовлетворять[**Code Conventions for the Java Programming Language**](http://java.sun.com/docs/codeconv/html/CodeConvTOC.doc.html)ипринципам**SOLID**
* В каждом классе должны быть описаны **конструкторы по умолчанию**, **конструкторы с параметрами**, инициализирующими поля классов, **методы получения** и **установки** значений в каждое из полей класса, метод **toString,** методы **hashCode** и **equals**

**Задание:**

Создайте класс Point в пакете model, описывающий точку в системе координат XOY, определите для него конструктор, необходимые арифметические операции в виде методов (сложения точек, умножения точки на число, деления точки на число). У класса Point два поля: x и y.

**Следующие задачи выполнять в виде static методов, расположенных в классе Point и принимающих на вход массив точек**

**A: Самая дальняя точка(maxDistPoint)**

Вычислить точку, наиболее удаленную от начала координат.

Для решения этой задачи напишите и используйтедополнительный метод **dist**, который возвращает расстояние от точки до начала координат. Если таких точек несколько, то вернуть последнюю, удовлетворяющую условиям задачи

**A1: Самые дальние точки(maxDistPoints)**

Вычислить точки, наиболее удаленные от начала координат.

**B: Центр масс(centerMass)**

Вычислите точку, являющуюся центром масс данного множества точек, т.е. точку, которая имеет в качестве своих координат среднее арифметическое по х и у.

**C: Диаметр множества(setDiam)**

Выведите диаметр данного множества – максимальное расстояние между двумя данными точками. Для решения этой задачи реализуйте и используйте перегрузку метода **dist**, который принимает на вход вторую точку и возвращает расстояние между двумя данными точками.

**D: Сортировка(sort)**

Определите для точек метод сравнения **compareTo**, реализовав интерфейс **Comparable**, сравнивающий точки по значению расстояния от начала координат. Отсортируйте данные точки в порядке возрастания расстояния от начала координат, используя метод sort из класса **Arrays**.

**E: Максимальный периметр(maxPerimetr)**

Среди данных точек найдите три точки, образующие треугольник с наибольшим периметром. Выведите данный периметр.

Для нахождения периметра треугольника напишите отдельныйметод **perimeter**, возвращающий периметр между тремя данными точками.

**F: Максимальная площадь(maxArea)**

Среди данных точек найдите три точки, образующие треугольник с наибольшей площадью. Выведите данную площадь.

Для нахождения площади треугольника напишите отдельный метод **area**.