Московский Государственный Университет

Отчет по задаче: «Поиск пары ближайших точек»

Камалов Руслан, гр.617

Москва, 2017

1. Постановка задачи

Пусть на плоскости даны n точек, заданные своими координатами, требуется найти среди них такие две точки, расстояние (в l_1 метрике) между которыми минимально. Тривиальный алгоритм — перебор всех пар и вычисление расстояния для каждой — работает за $O(n^2)$. В ходе выполнения работы реализован алгоритм, работающий за время $O(n \log n)$.

2. Описание метода решения

Алгоритм строится по общей схеме "разделяй-и-властвуй": алгоритм представлен в виде рекурсивной функции, которой передаётся множество точек; эта рекурсивная функция разбивает это множество пополам, вызывает себя рекурсивно от каждой половины, а затем выполняет операции по объединению ответов. Операция объединения заключается в обнаружении случаев, когда одна точка оптимального решения попала в одну половину, а другая точка — в другую.

Разбивать множество точек на два будем согласно их x-координатам: фактически мы проводим некоторую вертикальную прямую, разбивающую множество точек на два подмножества примерно одинаковых размеров. Такое разбиение удобно произвести следующим образом: отсортируем точки стандартно как пары чисел, т.е.:

$$(p = (x, y)) : p_i < p_j \iff (x_i < x_j) \vee (x_i = x_j) \wedge (y_i < y_j)$$

Возьмём среднюю после сортировки точку $p_m (m = \lfloor n/2 \rfloor)$, и все точки до неё и саму p_m отнесём к первой половине, а все точки после неё — ко второй половине:

$$A_1 = \{p_i \mid i = 0 \ldots m\}, \quad A_2 = \{p_i \mid i = m + 1 \ldots n - 1\}.$$

Сделав рекурсивный вызов от каждого из множеств A_1 и A_2, мы найдём ответы h_1 и h_2 для каждой из половинок. Возьмём лучший из них: $h = \min(h_1, h_2)$. После этого необходимо произвести стадию объединения, т.е. попытаться обнаружить такие пары точек, расстояние между которыми меньше h, причём одна точка лежит в A_1, а другая — в A_2. Очевидно, что для этого достаточно рассматривать только те точки, которые отстоят от вертикальной прямой раздела на расстояние, меньшее h, т.е. множество B рассматриваемых на этой стадии точек равно:

$$B = \{p_i \mid |x_i - x_m| < h\}.$$

Для каждой точки из множества B надо попытаться найти точки, находящиеся к ней ближе, чем h. Например, достаточно рассматривать только те точки, координата у которых отличается не более чем на h. Более того, не имеет смысла рассматривать те точки, у которых y-координата больше y-координаты текущей точки. Таким образом, для каждой точки p_i определим множество рассматриваемых точек $C(p_i)$ следующим образом:

$$C(p_i) = \{p_j \mid p_j \in B, \quad y_i - h < y_j$$

В отсортированном множестве B по y-координате найти множество $C(p_i)$ равносильно взятию несколько точек подряд до точки p_i. Итак, в новых обозначениях стадия объединения выглядит следующим образом: построить множество B, отсортировать в нём точки по y-координате, затем для каждой точки $p_i \in B$ рассмотреть все точки $p_j \in C(p_i)$, и каждой пары (p_i, p_j) посчитать расстояние и сравнить с текущим наилучшим расстоянием.

3. Инструкция по работе с программой

К коду на C++ прилагается exe-модуль. Параметром запуска программы является имя файла с точками.