

Отчёт о выполнении лабораторной работы №1 по курсу «Алгоритмика».

Мурат Апишев, группа 617 (ММП ВМК МГУ).

1 Постановка задачи

В данной работе предлагается разработать и реализовать программно алгоритм поиска на множестве Q , состоящем из $n \geq 2$ точек, пары точек, расстояние между которыми минимально. Расстояние предлагается измерять по евклидовой метрике. Предполагается, что две точки могут совпадать, в этом случае от программы ожидается выдача всех подобных точек. Программа должна реализовывать алгоритм «разделяй-и-властвуй», и иметь временную сложность $O(n \log(n))$.

2 Метод решения

Для решения поставленной задачи были реализованы следующие функции:

- Основные:
 1. Функция сортировки слиянием массива.
 2. Функция поиска пар точек с минимальным расстоянием.
- Вспомогательные:
 1. Процедура чтения данных из файла.
 2. Функция подсчёта евклидового расстояния между парой точек.
 3. Процедура печати результатов.
 4. Функция поиска пар точек с минимальным расстоянием на множестве с помощью полного перебора.

2.1 Сортировка массива слиянием

Алгоритм выполняется в два этапа — на первом выполняется рекурсивный запуск на двух половинах входного массива, базой рекурсии является возврат массива размера 1.

Второй этап заключается в комбинировании результатов рекурсивного запуска для текущего полученного массива. Для этого элементы массивов просматриваются параллельно и выстраиваются в итоговом массиве в правильном порядке. Если какой-то из двух просматриваемых массивов оказался короче, элементы второго корректно дописываются в конец.

Функция, реализующая алгоритм, получает на вход итератор на начало сортируемого массива, итератор на начало временного массива аналогичной длины, и два целых индекса левой и правой границ текущего сортируемого подмассива.

2.2 Поиск пар точек с минимальным расстоянием методом «разделяй-и-властвуй»

Прежде всего массив точек сортируется по первой и второй координате, т.е. на выходе получаются два массива X и Y . При этом для обоих массивов сохраняется индексация элементов в X .

Принцип этого алгоритма схож с описанным выше. Аналогично рассматриваются два этапа, в первом производится рекурсивный запуск алгоритма на левой и правой половинах входного массива, базой рекурсии является полный перебор на входном массиве с длиной менее 4-х элементов.

В обоих полученных после рекурсивного запуска массивах найдены пары точек с минимальным расстоянием, перед комбинированием производится слияние результатов: если минимальное расстояние для обоих подмассивов одинаково, то их результаты объединяются; в противном случае результатом признаются пары точек того подмассива, для которого минимальное расстояние меньше.

Все описанные выше операции производятся над массивом точек X .

Этап комбинирования результатов заключается в следующем. В цикле по элементам полного массива, отсортированного по второй координате (Y), производится проверка на попадание точки в δ -полосу, если попадает — добавляем такую точку в новый массив Y_δ .

Затем в цикле Y_δ производится проверка на то, что первая координата элемента находится в рассматриваемом на данном этапе подмассиве массива X . Если это не так, то точка пропускается. Если так, то выполняется просмотр шести её соседей по возрастанию в Y_δ . Если кто-то из соседей находится с рассматриваемой точкой в одной половине текущей рассматриваемой части X , то такой сосед игнорируется (он уже был учтён в рекурсивном вызове). В противном случае между точками считается расстояние, и если оно меньше текущего минимального, полученного по результатам рекурсивных вызовов, общий результат обновляется.

Функция, реализующая алгоритм, получает на вход итератор на начало сортируемого массива X , константную ссылку на массив Y , и два целых индекса левой и правой границ текущего сортируемого подмассива X .

3 Инструкция по запуску программы

Запуск исполняемого файла производится следующей командой:

```
apishev_lab_1.exe <data_path>,
```

где `<data_path>` — полный путь к файлу с данными. Файл может иметь один из двух форматов: однострочный (в этом случае первое значение интерпретируется как количество точек, а каждая пара следующих — как первая и вторая координата точки соответственно), либо многострочный (в первой строке находится одно целое число — количество точек, дальше в каждой строке идёт пара координат одной точки).