**深 圳 大 学 实 验 报 告**

**课程名称： 算法设计**

**实验名称： 分治法求最近点对问题**

**学院：计算机与软件学院 专业： 计算机科学与技术（创新班）**

**报告人： 何泽锋 学号： 2022150221 班级： 高性能特色班**

**指导教师： 杨烜**

**实验时间： 2024年4月12日**

**实验报告提交时间： 2024年4月12日**

**教务处制**

**一．实验目的**

1.掌握分治法思想。

2.学会最近点对问题求解方法。

**二．问题描述**

1.使用暴力法对N=100000-1000000的点对进行比较，得到最近的点对

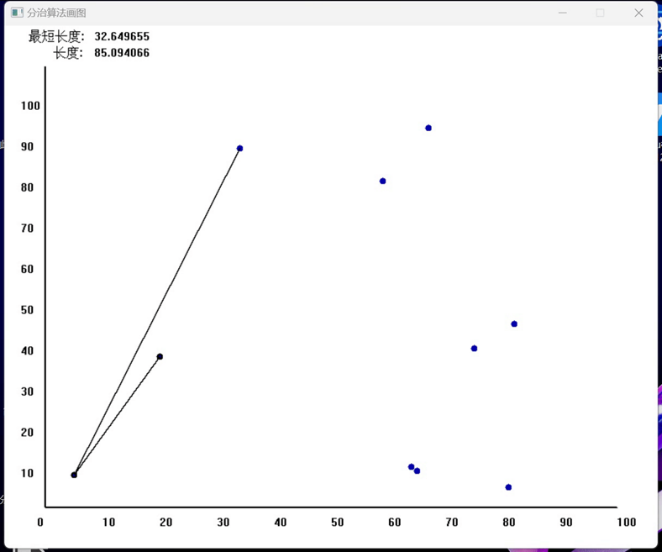
2.使用分治法对N=100000-1000000的点对进行比较，得到最近的点对

3.图形化输出比较过程

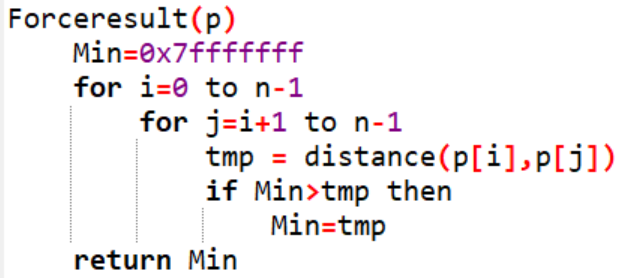
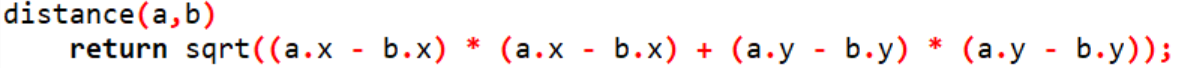
**三．算法原理**

**1.暴力法**

①原理：使用两层循环，遍历任意两点距离，记录两点的最小距离，若出现比当前最小距离小则更新最小距离

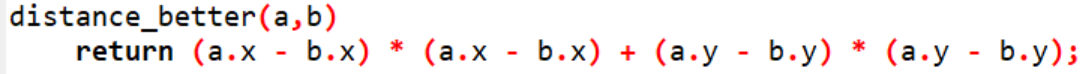


②核心伪代码：



③细节解释：源代码使用时需要注意distance的返回值为double，要注意数据类型，防止精度丢失

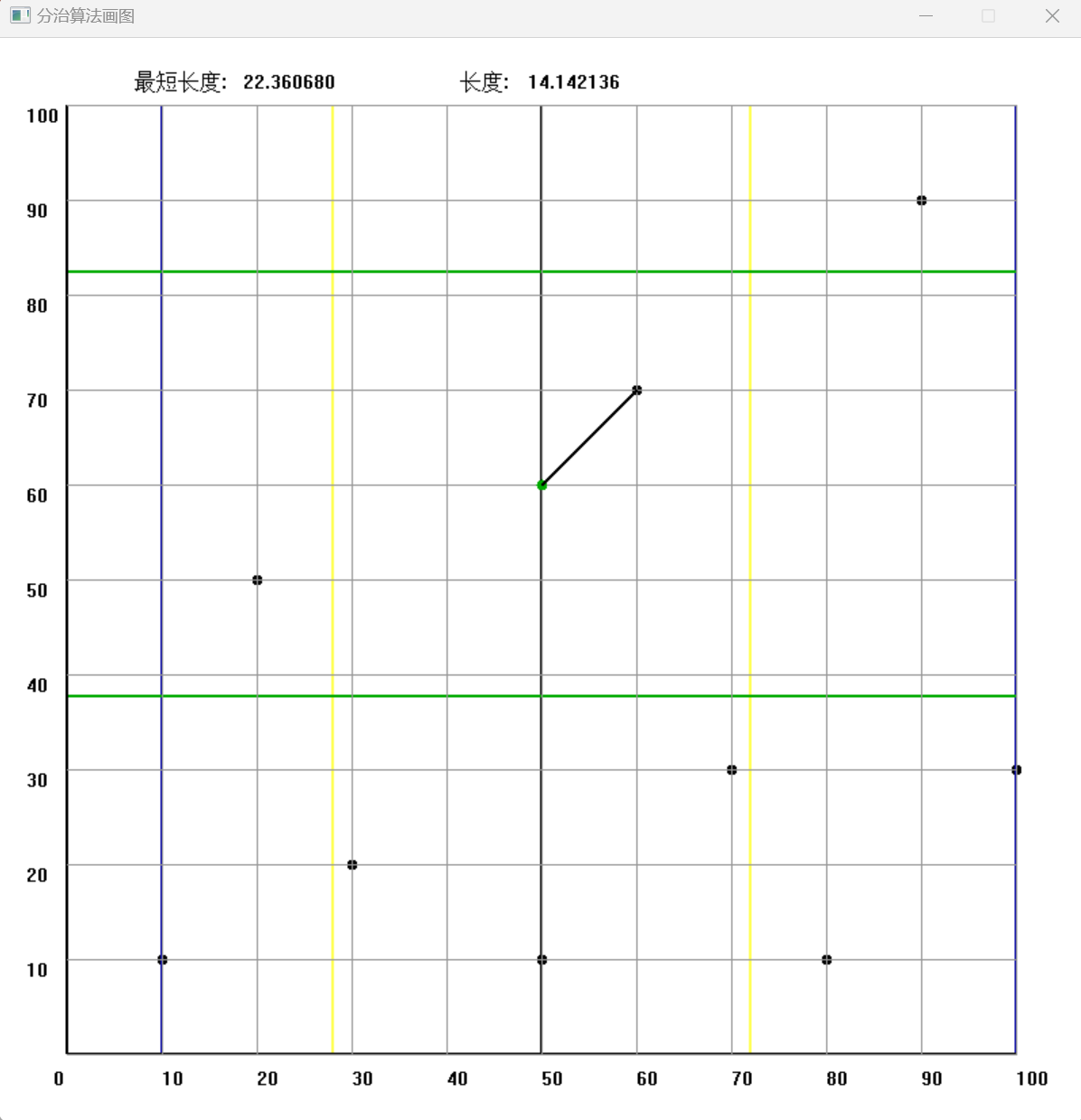
④优化：对distance进行优化，在计算距离时不再对每一次的结果都开根号，保留平方数值进行比较，最终结果再开根，可以减少一定的操作次数，减少耗时



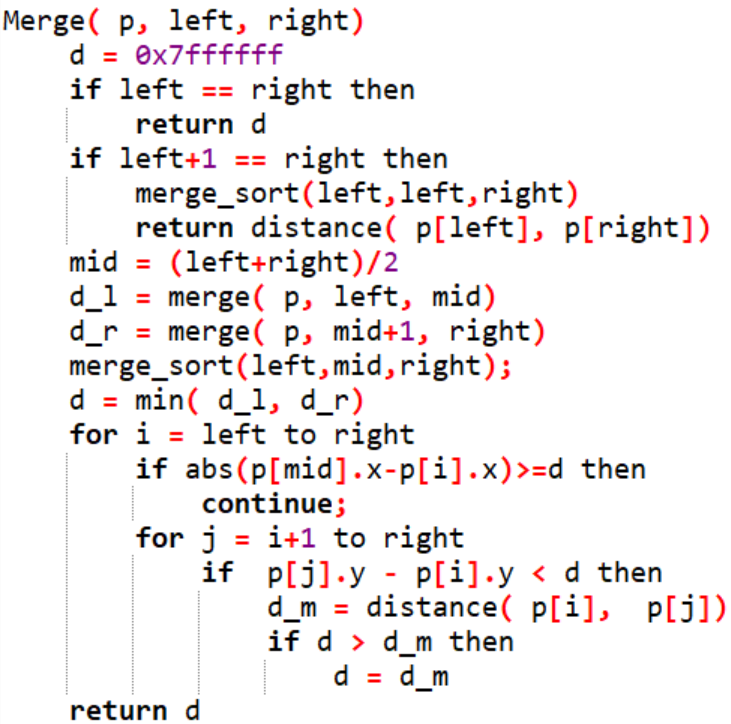
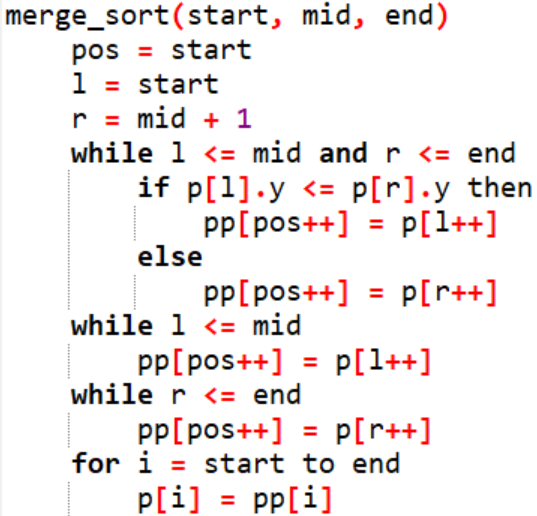
**2.分治法**

①原理：按照x从小到大对点排序，二分寻找最近点对，合并时从中间遍历距离已知最短的范围内的点，再通过两两比较的方式在中间范围内尝试寻找更小的距离

如下图所示，mid=50，递归遍历左右两侧后可以得到当前最小的d，在图上表示为黄色线，使用归并排序对黄线范围内的点按照y值从小到大排序。依次遍历黄线范围内的点与其他范围内的点的距离，此处还使用了绿线，绿线为当前遍历的点上线距离为d的范围，若两点都在黄线范围内，当在绿色范围外，说明二者之间的距离也就大于d无需继续判断，此过程时间复杂度为O(n)，后续通过鸽巢原理证明



②核心伪代码：

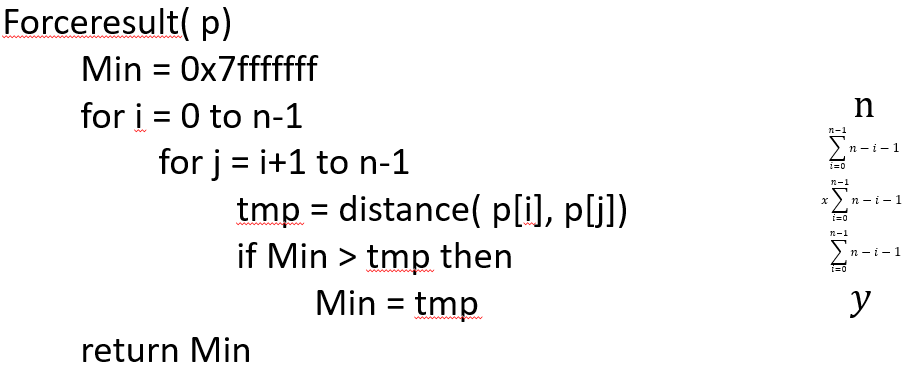
 

③细节解释：实验采用的是归并法对(midd)范围内的点进行y值排序，并通过鸽巢原理将合并代价降低至O(n)

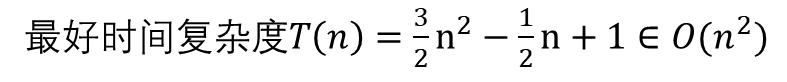
④优化：实验运用的是优化的代码，即使用归并排序在每一趟对y值进行排序。未优化的代码则是通过快速排序对(midd)范围内的点进行y值排序。二者的时间分别为O(n)和O(nlogn)，具体复杂度计算会在下方讲述

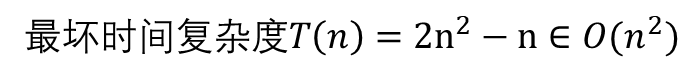
**四．算法效率分析**

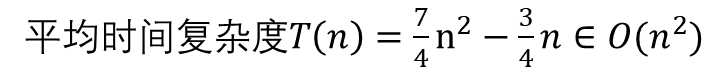
**1.暴力法效率分析：**

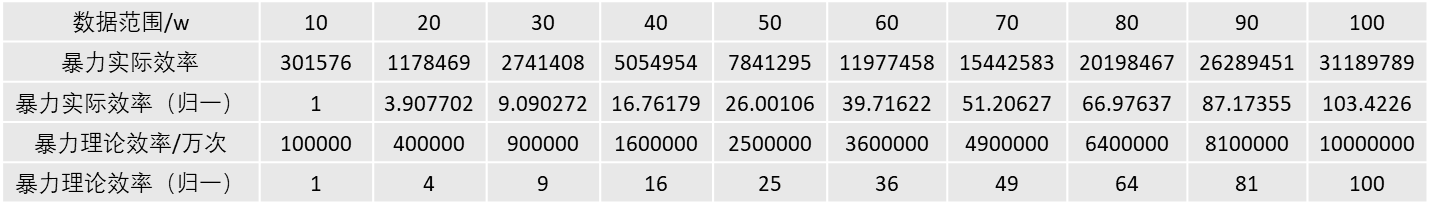


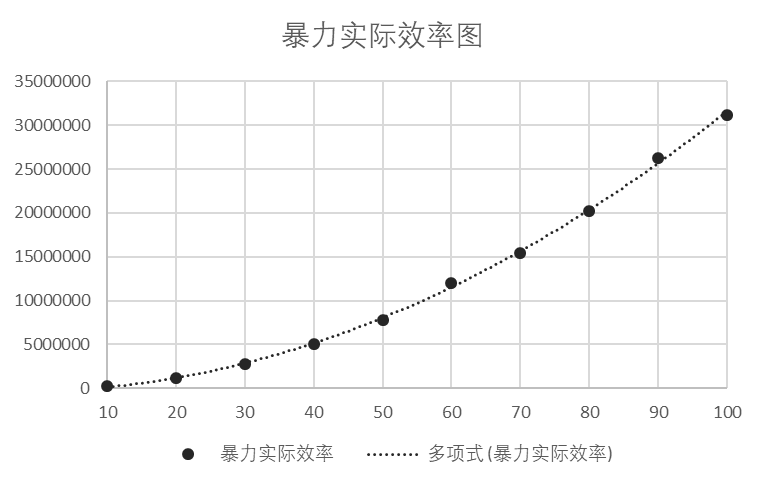
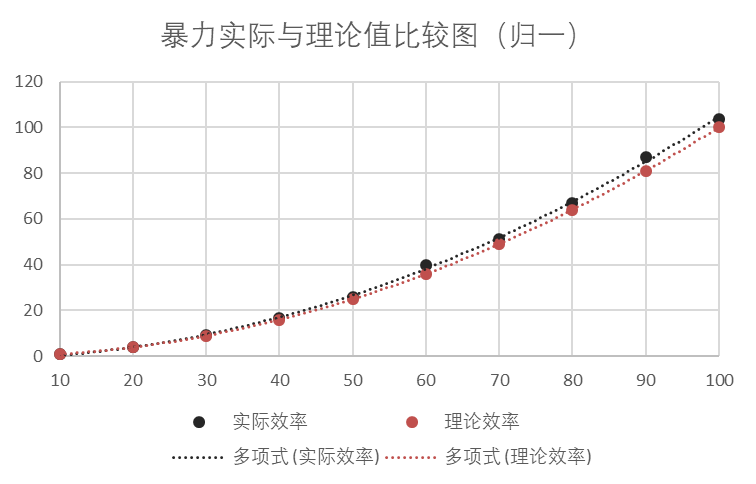
x求距离的函数操作数得到，但sqrt是由二分实现，具体操作数不好计算，但x的值不会影响最终时间复杂度，此处令x=1



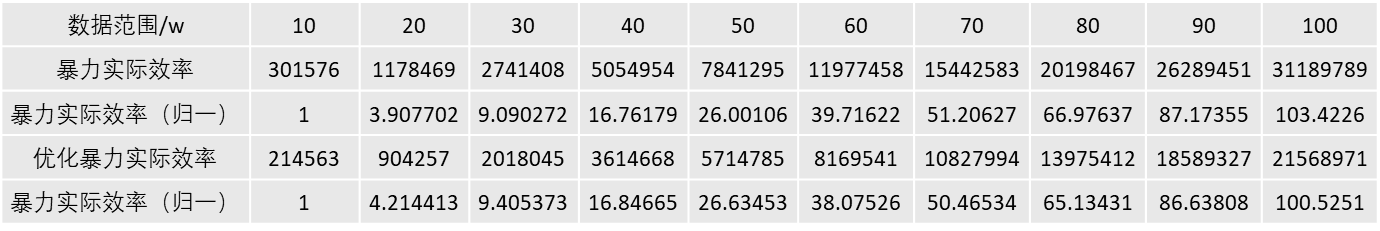


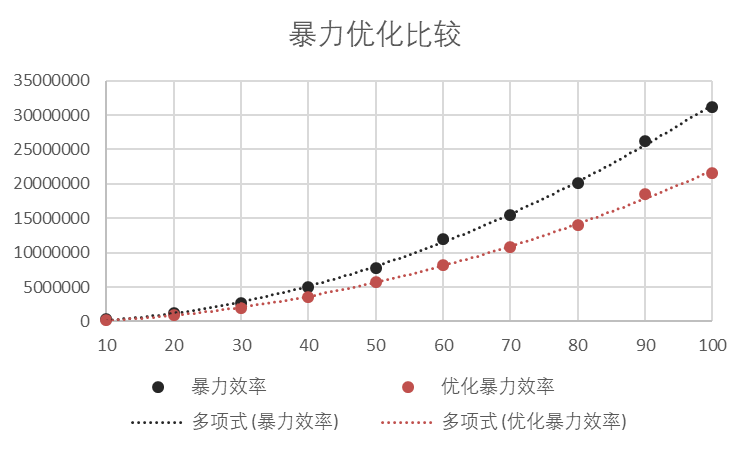




优化后效率比较：

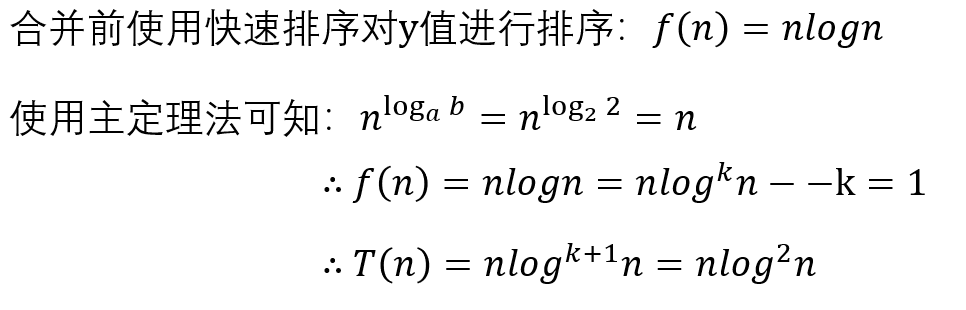




**2.分治法效率分析：**

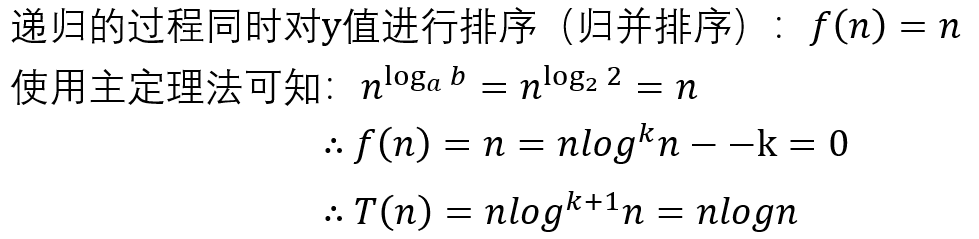
①快速排序：

快排

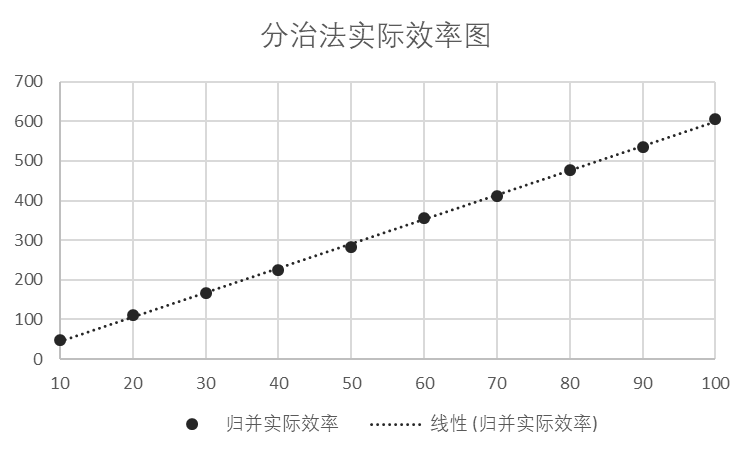
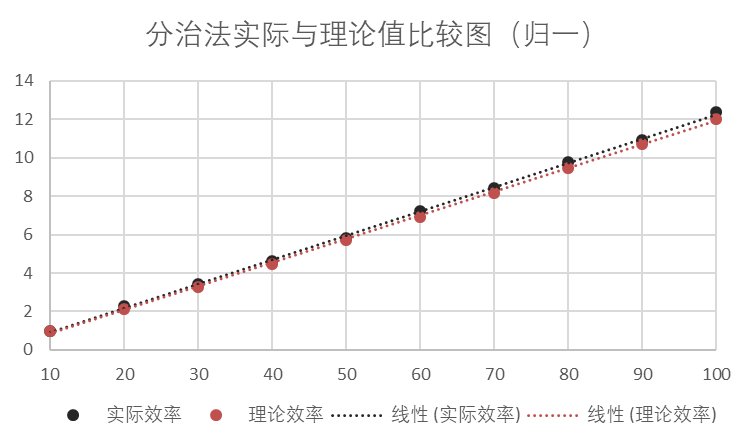


②归并排序：

归并

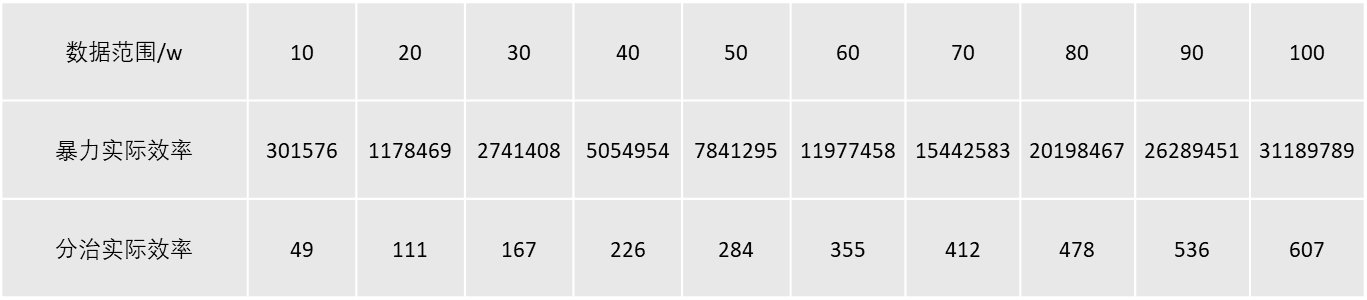


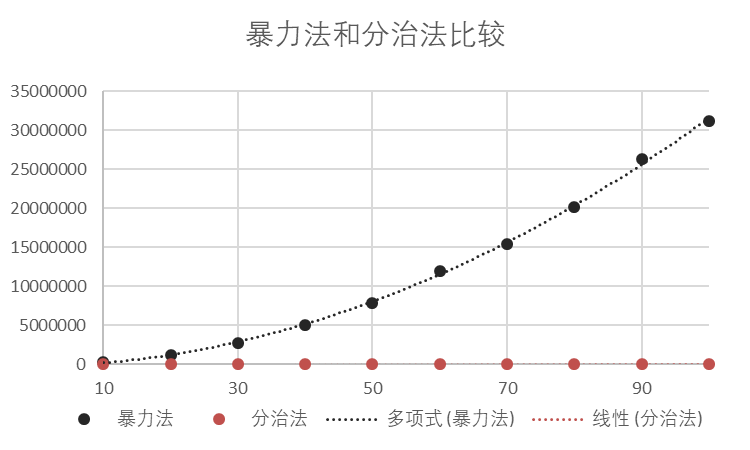


**3.暴力法与分治法效率比较：**

由数据可知二者数量级是完全不一样的，在10w-100w的数据范围内耗时相差非常大，暴力法消耗的时间远大于分治法。



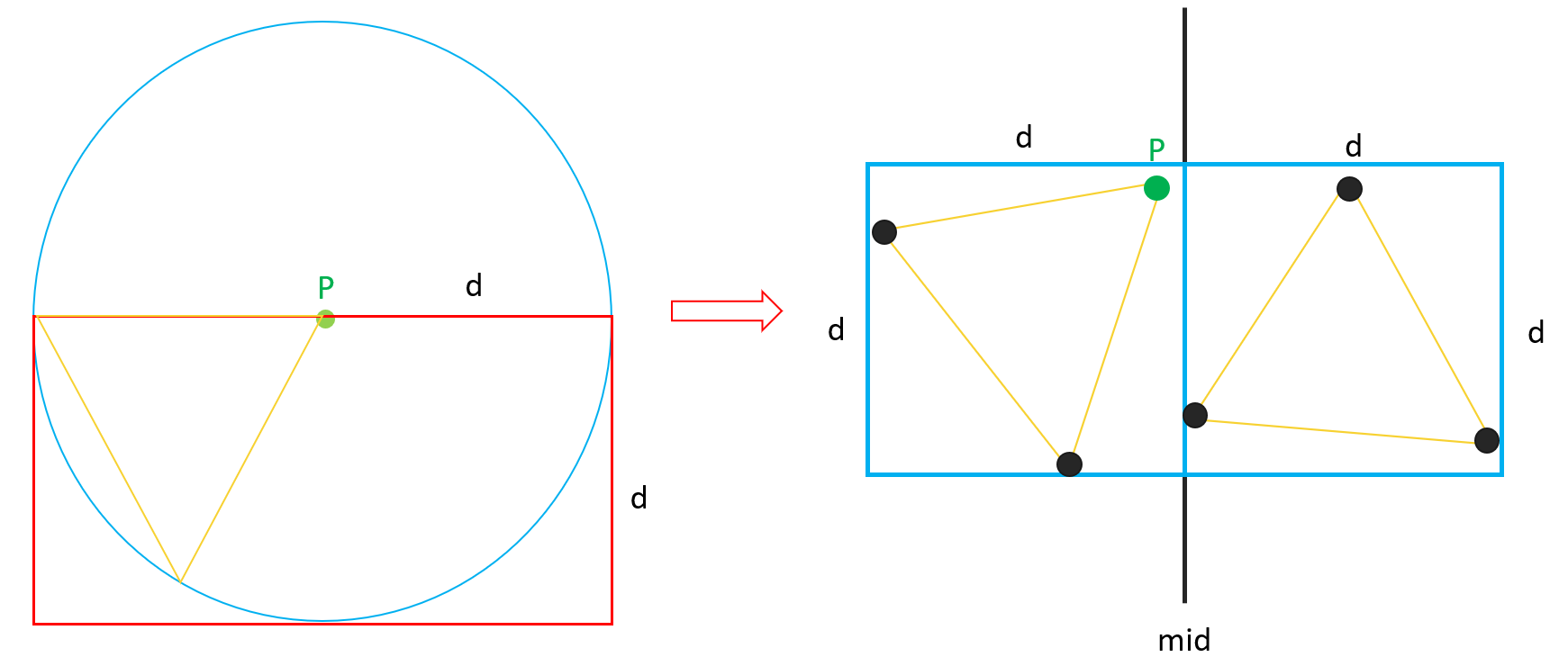


**五．细节展示**

**1.鸽巢原理：**

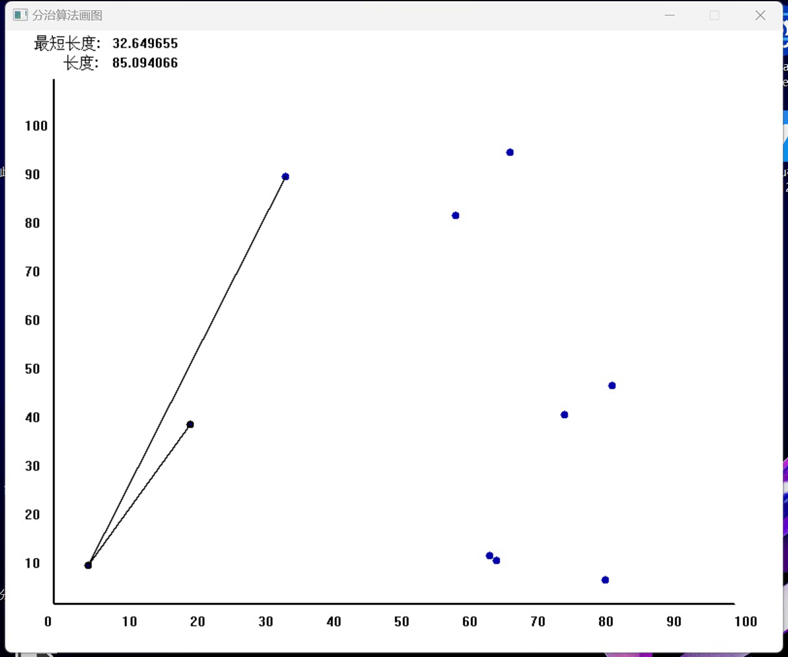
对于合并时的点来说，若在同一侧，则二者距离一定≥d，因此要出现小于d的距离则两点一定在mid两则，具体来说两侧的点之间要比较多少次呢？

假设点p在mid线上，可以画出如图所示的半径为d的圆，分析其下半部分（上半部分可以归到其他点的范围），对圆进行一定的放缩，可以得到d\*2d的矩形，已知圆上的点到圆心距离一定为d，因此对于在一侧的点在这个矩形范围内只能出现三角形分布，且三角形的边大于等于d，对于一个点若存在一个于其距离小于d的点，二者一定分布在两侧，可以做出最差情况图像如图所示，一个点最多只会与五个点进行比较因此对于每个点的比较复杂度为O(5)，一共需要比较n个点，总比较时间复杂度可以优化至O(n)

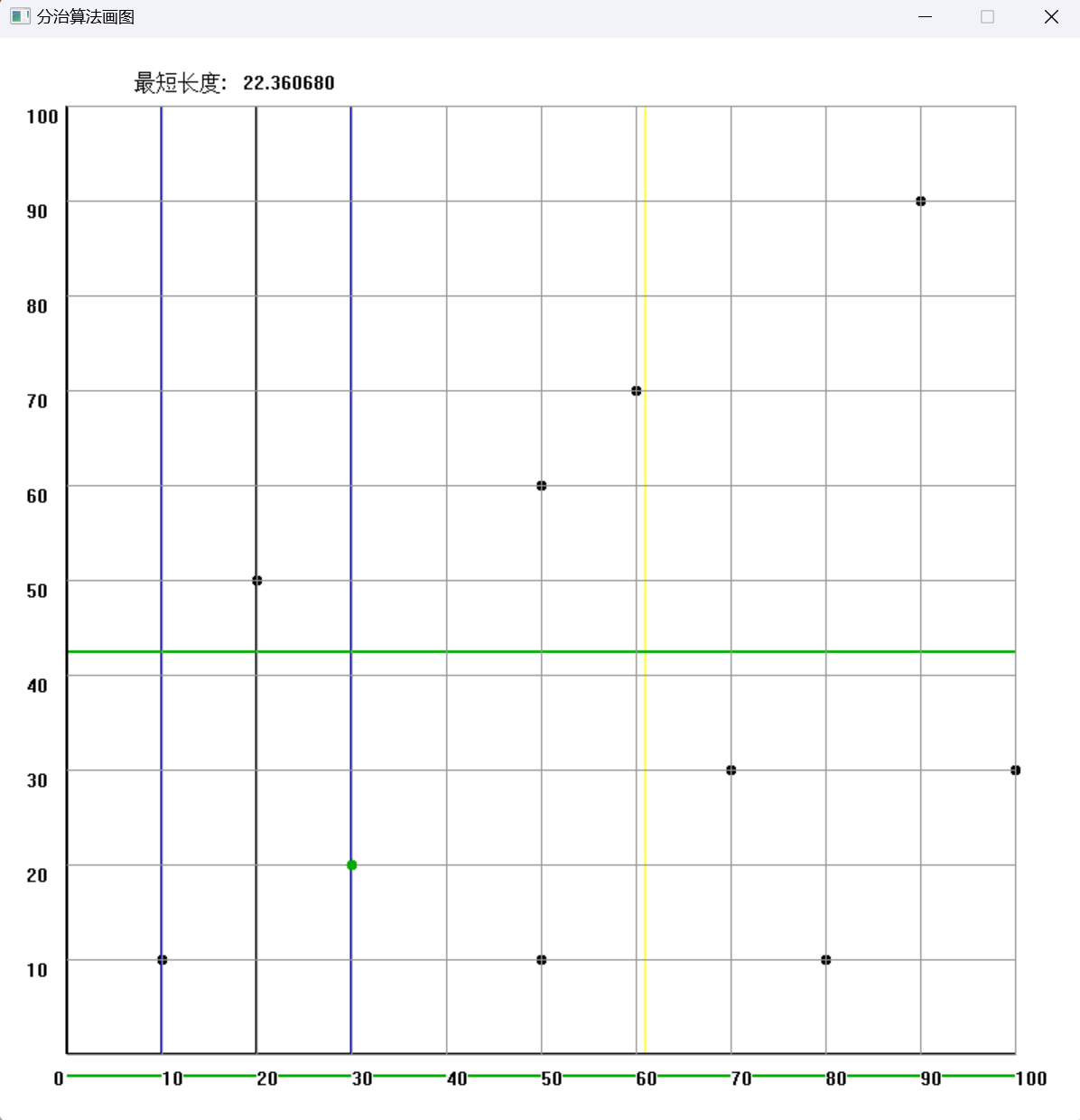
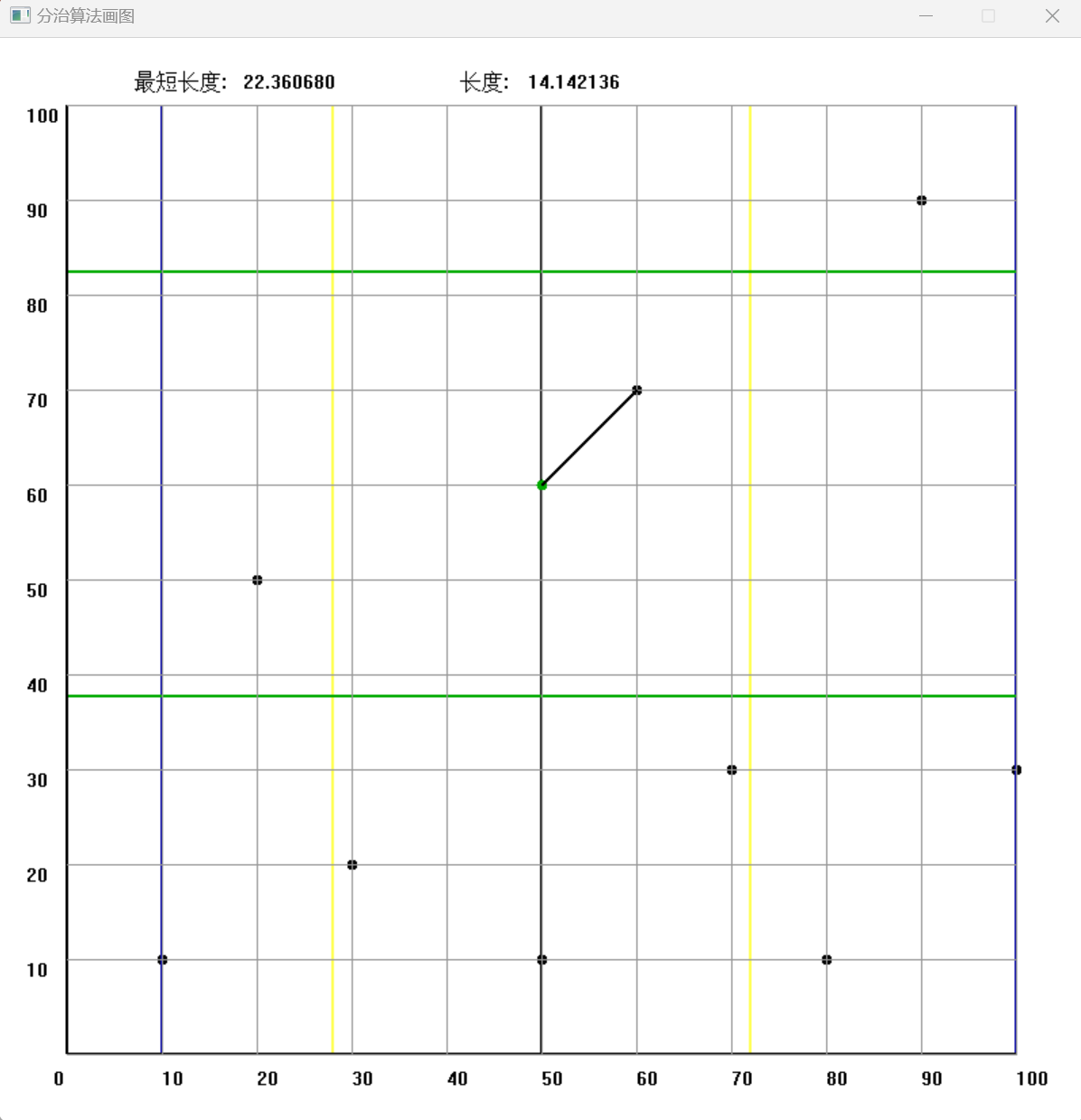


**2.图形化展示：**

①暴力法：依次比较两点之间的距离，若当前距离小于已知的最小距离，则更新最小距离



②分治法：图中蓝线为left和right边界，黑线为mid，黄线为midd，绿色的点为当前比较的点，绿色的线为当前比较点的y值d

**六．实验结论**

1.实验包含两种方法，分别是暴力法和分治法，时间复杂度分别为和

2.实验过程对暴力法进行了简单的优化，通过去除每次计算距离时的开根号操作，减少操作次数，从而提高了效率，减少运行时间，但并没有降低代码的时间复杂度。

3.分治法采用的也是优化的代码，首先合并前对y值进行排序，排序后可以通过鸽巢原理将比较的时间复杂度优化到,否则对合并时的点两两比较复杂度会达到。而对于y值排序也进行了优化，若采用快速排序，时间复杂度会达到,最终代码的复杂度会到达，而通过归并排序，可以在分治的同时对数据进行分割，从而对于每次合并其时间复杂度为，最终代码的时间复杂度为

4.实验通过图形化展示暴力法、分治法的比较过程，但c++并不包含画图的库，通过添加easyx库，可以实现基本的画图。

|  |
| --- |
| 指导教师批阅意见：    成绩评定：  指导教师签字：  年 月 日 |
| 备注： |

注：1、报告内的项目或内容设置，可根据实际情况加以调整和补充。

2、教师批改学生实验报告时间应在学生提交实验报告时间后10日内。