**高级算法设计与分析 课程**

**实验报告**

**实验一：实验题目**

姓名：？？？

学号：？？？

班级：？？？

评分表：（由老师填写）

|  |  |
| --- | --- |
| 最终得分： | |
| 对实验题目的理解是否透彻： | |
| 实验步骤是否完整、可信 ： | |
| 代码质量 ： | |
| 实验报告是否规范 ： | |
| 趣味性、难度加分 ： | |
| 特 色： | 1 |
| 2 |
| 3 |

**一、实验题目概述**

求解凸包问题：输入是平面上n个点的集合Q，凸包问题是要输出一个Q的 凸包。其中，Q的凸包是一个凸多边形P，Q中的点或者在P上或者在P中。

**二、对实验步骤的详细阐述**

主要分为三个小的实验：

实验一——基于枚举的实验：考虑Q中的任意四个点A、B、C、D，如果A处于BCD构成的三角 形内部，那么A一定不属于凸包P的顶点集合。对于n个节点，每次选取4个，一个具有中选择，因此最终的时间复杂度为 m.

实验二——使用Graham-Scan的方法。主要包含以下步骤：

1. 从点集中找到一个基准点，通常选择y坐标最小的点，如果有多个，则取其中x坐标最小的点。
2. 以基准点为参考，按照每个点相对于基准点的极角大小对所有点进行排序。如果两个点有相同的极角，则距离基准点较近的点排在前面。
3. 初始化一个栈，将排序后的前两个点和基准点按顺序压入栈中。
4. 从排序后的剩余点集中依次考虑每个点，并与栈顶的两个点一起判断构成的转折方向。如果是逆时针转向（即形成左转），则将当前点压入栈中；如果是顺时针转向（即形成右转），则将栈顶的点弹出，直到栈顶的两个点与当前点构成逆时针方向为止。
5. 当所有点都被考虑过后，栈中剩余的点就构成了点集的凸包。

这个算法的时间复杂度是O(n log n)，主要耗时在于点的排序过程。凸包的构建过程本身是O(n)。

实验三——基于分治的算法。主要包含以下步骤：

1. Divide。将点集按照横坐标排序，并等分为两个部分。如果点集的数量小于4就可以认为这个点集构成一个凸包。
2. Conquer。现在已经得到了左右两个凸包，考虑如何将两个凸包合并为一个。以左边凸包的内点（也可以为一个指定的节点）为基准点，找到右凸包关于此节点的极角最大和最小的节点，这两个节点可以将右凸包分为左右两个部分。使用类似于归并排序的Merge方法，可以将左凸包、右边的左和右的两个部分按照此基准点进行排序，案后按照Graham-Scan的方法来得到凸包。

总的递归式为,根据Master定理，可以得到最终的时间复杂度为.

**三、实验数据**

**1. 实验设置**

**实验环境**：

Windows 11，vscode，python

**数据**：

使用生成的1000-5000的在[0,0]-[100,100]矩形范围内的点。

**2. 实验结果**

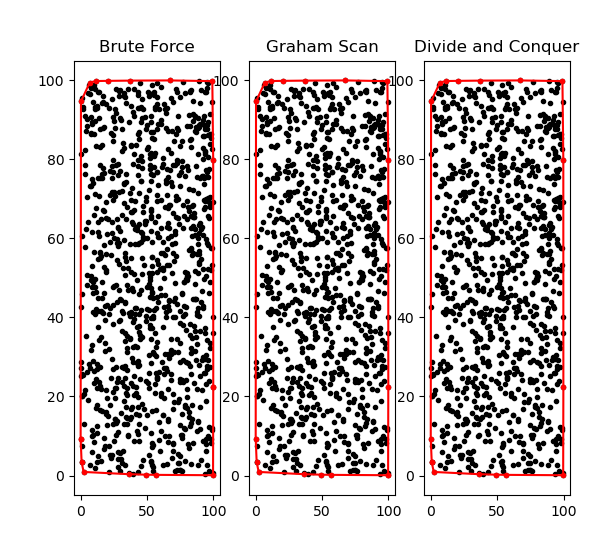


图 1 实验结果

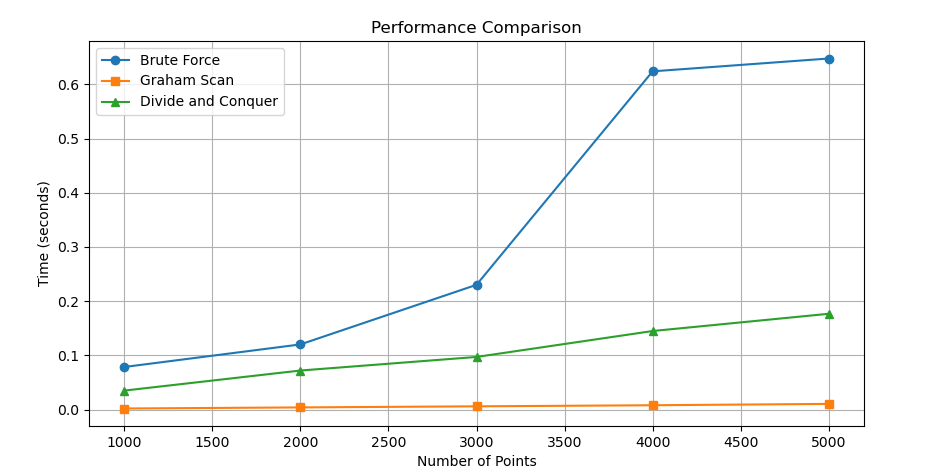


图 2 运行速度

**四、对实验结果的理解和分析**

1. 三种算法都能得到正确的结果。
2. 暴力枚举具有的时间复杂度，而分支和Graham-Scan具有的时间复杂度，并且从上图观察，Graham-Scan具有更小的常数。

**五、实验过程中最值得说起的几个方面**

通过明确且精确地编写代码，加深了对分治算法的理解，并掌握了分治算法的设计原则与实现方法。利用高级编程语言成功实现了分治算法。通过研究凸包问题的求解方法，包括直观的算法与基于分治策略的算法，理解了算法复杂度的概念以及不同解决方案的优劣。实验表明，在小规模数据集上，不同算法之间的性能差异不显著；然而，随着数据量的增加，算法的时间复杂度成为性能的决定性因素。