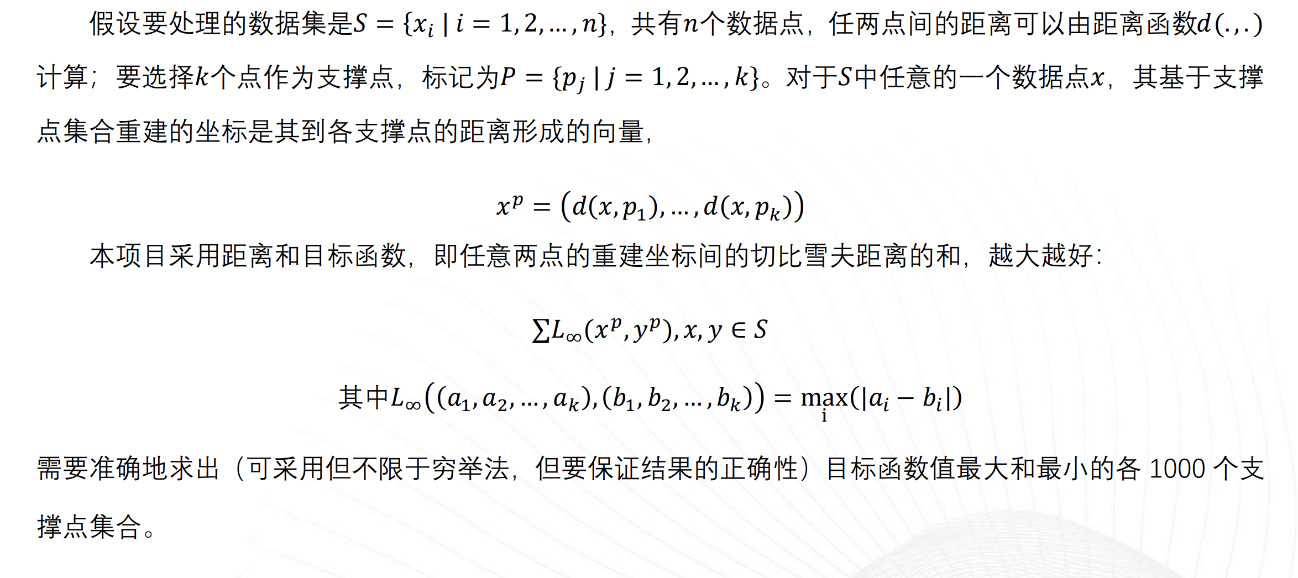
多核平台下的并行计算课程实验指导书

**[实验目的]**：利用所学的并行技术，实现支撑点选择算法的并行化和优化，提高数据处理的效率，并通过实验充分理解与掌握并行计算技术。

**[实验安排]**：实验共分 16次，其中第1次实验为熟悉 Linux 环境，安装并行编程软件，并运行已有的串行代码；接下来的 15 次实验，使用任选一种并行编程模型：包括课堂上讲过的 MPI、OpenMP、Pthread 或者课堂上没有讲过但通过自学掌握的其它工具如：Intel Cilk，Intel TBB等实现多核CPU处理器上并行化，如有余力可以在算法和粗细粒度上进一步优化代码。所有工作要在 15 次实验课内完成（即30个课时）。

**[题目简介]：**

**[优化策略提示]：**

1. **可以选用多线程或者多进程的并行方式优化计算热点。**
2. **另外鼓励从算法上进行进一步优化，比如使用高效算法，等效修改任务划分方式以实现更好的任务并行度等。**
3. **建议尝试使用向量化等细粒度优化技术。**

**[使用方法]：**

1. 源代码包括以下文件

a) pivot.c 源代码文件

b) uniformvector-2dim-5h.txt 为第一组输入数据文件；uniformvector-4dim-1h.txt为第二组输入文件（运行时间较长）。

c) refer-2dim-5h.txt 基准输出文件

2. 程序使用方法：

a) 参考编译命令：gcc pivot.c -lm -o pivot

b) ./pivot

3. 考核程序求出目标函数值最大和最小的各 1000 个支撑点组合所用总时间，以程序输出“Using time”时间为准，不包括读写文件的时间，不得修改计时函数的位置。

4. 不得修改输入文件。

5. 可以改变源代码的数据结构和数据类型。

6. 以 result.txt 结果文件作为评判标准，所选出的目标函数值最大和最小的各 1000 个支撑点集合及其顺序须与 refer 基准文件(refer-2dim-5h.txt和refer-4dim-1h.txt)完全相同。即先按目标函数值降序排序，输出目标函数值前 1000 大的点集；再按目标函数值升序排序，输出目标函数值前 1000 小的点集。

7. 可自行更改编译方式和编译参数，但需要留存脚本文件或 Makefile 文件。

8. 测试环境为Intel 铂金系列处理器，编译器为GCC8.3或Intel OneAPI。

**[实验分组]**：实验可以单独完成也可以分组完成，但每小组最多3人。

**[实验报告]**：实验完全结束后每人需要提交一份实验结果zip格式压缩包（先放到文件夹中再压缩，不要提交输入、输出和验证数据集，不要使用rar压缩格式），压缩包需要包括一份实验报告（.doc 后缀）、源程序和readme文件（内含执行所实现的并行程序需要的参数说明比如：如何设定程序运行时的线程数等）。其中实验报告需要包括并行算法的设计思路、优化技术以及使用不同线程数目时的运行时间与加速比情况，具体格式与内容请参见附1。压缩包的名字请写成：**班级+PC+学号+姓名** 格式，并发送到SDUHPC2016@163.com，邮件主题请标明：“**并行计算实验：班级+PC+学号+姓名**”。

附1

多核平台下的并行计算课程实验报告

姓名： 学号： 班级

1. **填写下表，给出所使用的软硬件环境参数**

|  |  |
| --- | --- |
| 操作系统版本 |  |
| 并行编程模型 |  |
| 编译器版本 |  |
| CPU型号、主频及核数 |  |
| 内存型号，大小及主频 |  |

1. **填写下表，给出程序使用不同线程数目时的运行时间与加速比（也可以自己设计图表描述不同线程数时程序的运行时间及加速比）**

|  |  |  |
| --- | --- | --- |
| 线程数目 | 运行时间（秒） | 加速比 |
| 1 |  | 1.0 |
| 2 |  |  |
| 4 |  |  |
| 8 |  |  |
| 16 |  |  |
| …… | …… | …… |

1. **并行算法设计和优化思路**