**中山大学计算机院本科生实验报告**

**（2024学年春季学期）**

课程名称：并行程序设计  **批改人：**

|  |  |  |  |
| --- | --- | --- | --- |
| 实验 | 6-Pthreads并行构造 | 专业（方向） | **计算机科学与技术** |
| 学号 | **21307174** | 姓名 | **刘俊杰** |
| Email | **liujj255@mail2.sysu.edu.cn** | 完成日期 | **2024/5/3** |

# 实验目的

parallel\_for并行应用

使用此前构造的parallel\_for并行结构，将heated\_plate\_openmp改造为基于Pthreads的并行应用。

**heated plate问题描述：**规则网格上的热传导模拟，其具体过程为每次循环中通过对邻域内热量平均模拟热传导过程，即：

，

其OpenMP实现见课程资料中的heated\_plate\_openmp.c。

**要求：**使用此前构造的parallel\_for并行结构，将heated\_plate\_openmp实现改造为基于Pthreads的并行应用。测试不同线程、调度方式下的程序并行性能，并与原始heated\_plate\_openmp.c实现对比。

# 实验过程和核心代码

2.1 实验思路

## ①首先使用实验5实现的parallel\_for来模拟实现openmp的加速功能

## 2.2 parallel.c

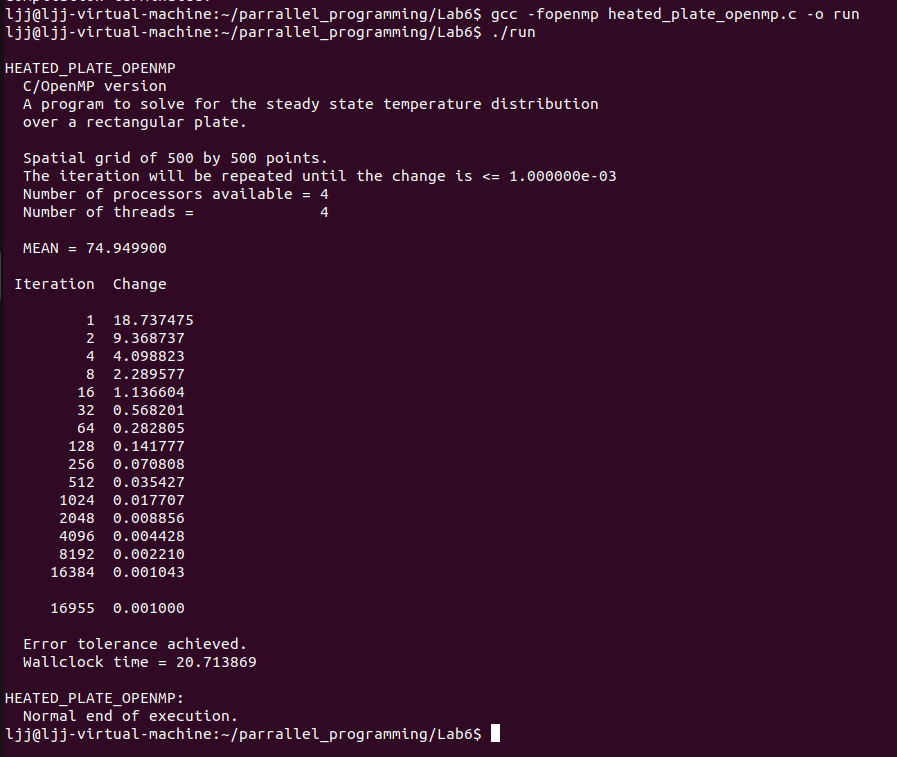
2.3 pthread版本:heat\_plate\_pthread.c

# 实验结果

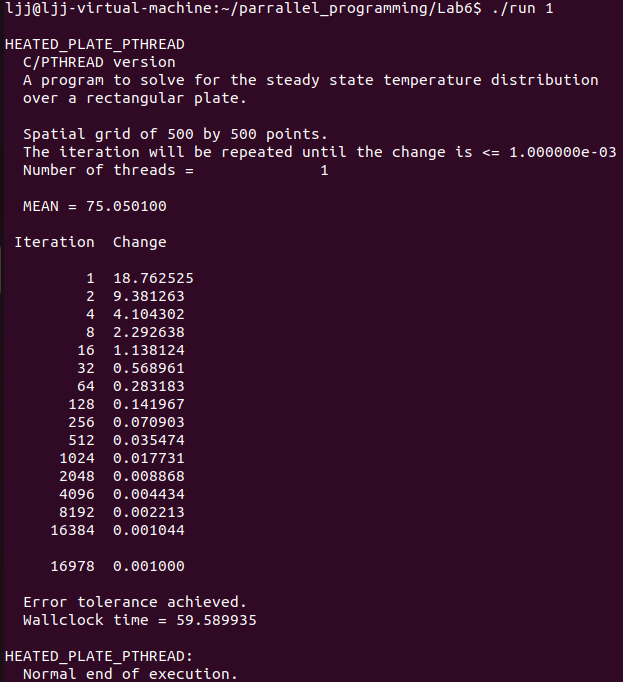
3.1 head\_plate\_openmp.c

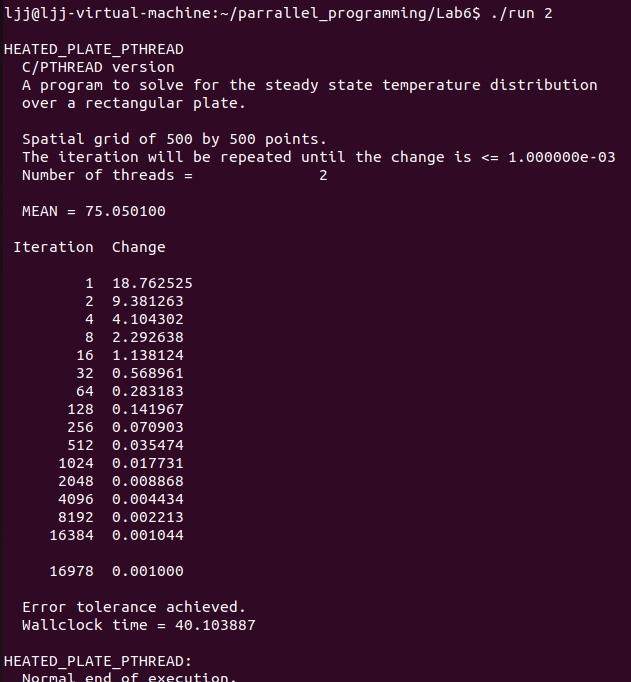
## openmp版本的实验结果:

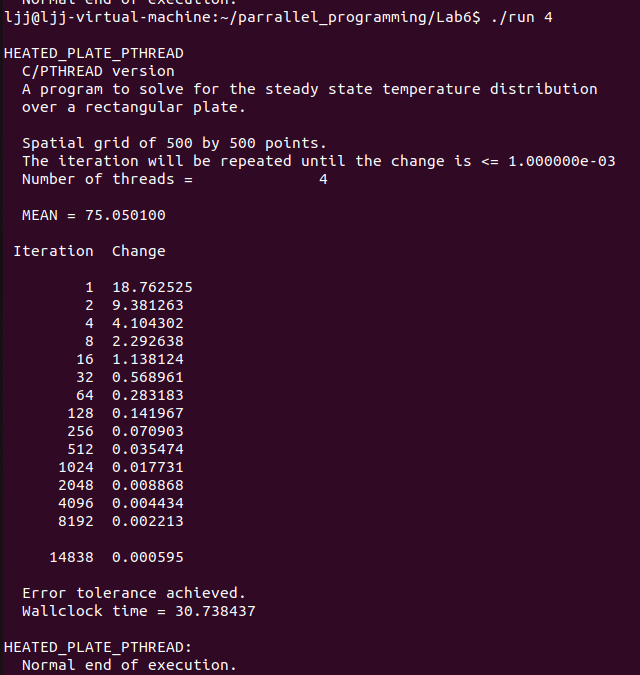
针对500X500的矩阵,用四个线程的加速结果:

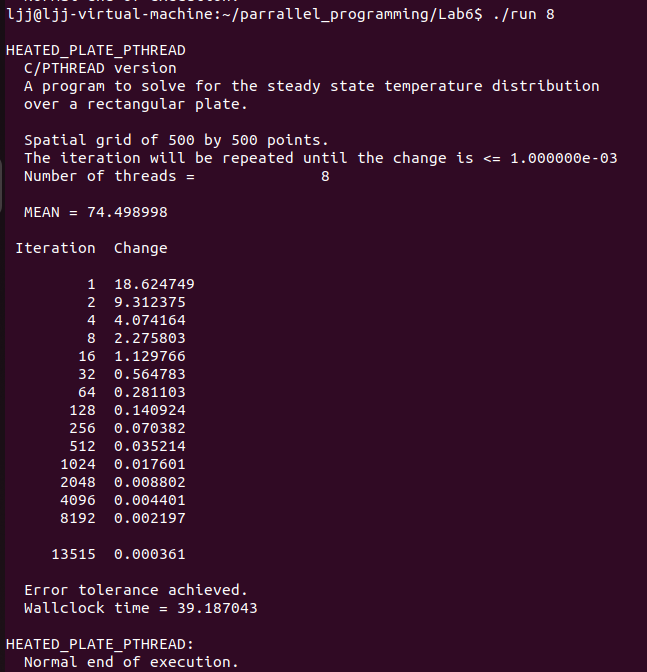


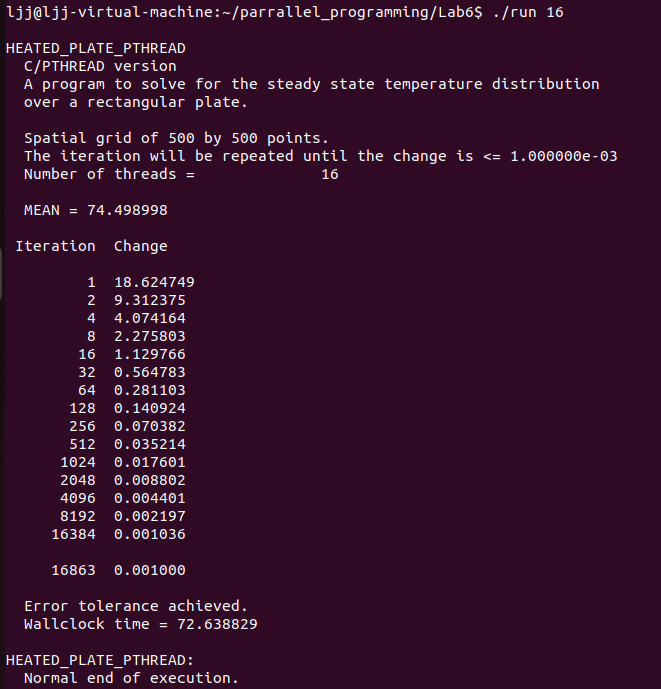
## 3.2

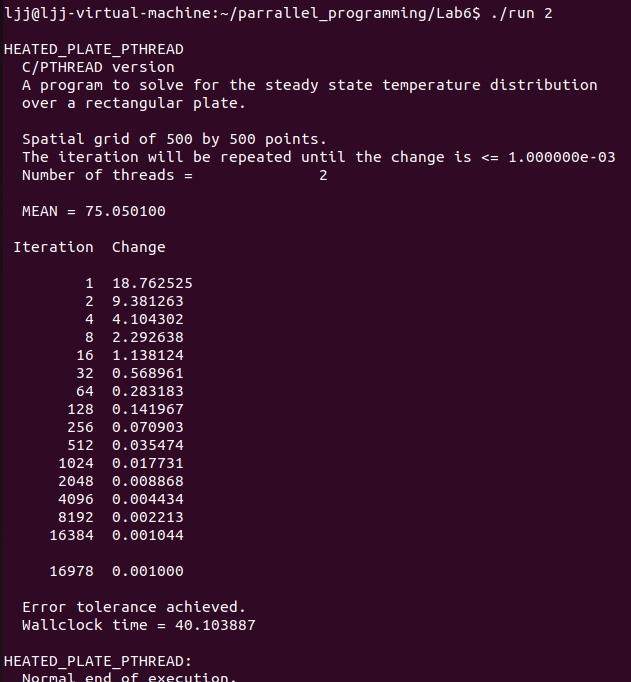




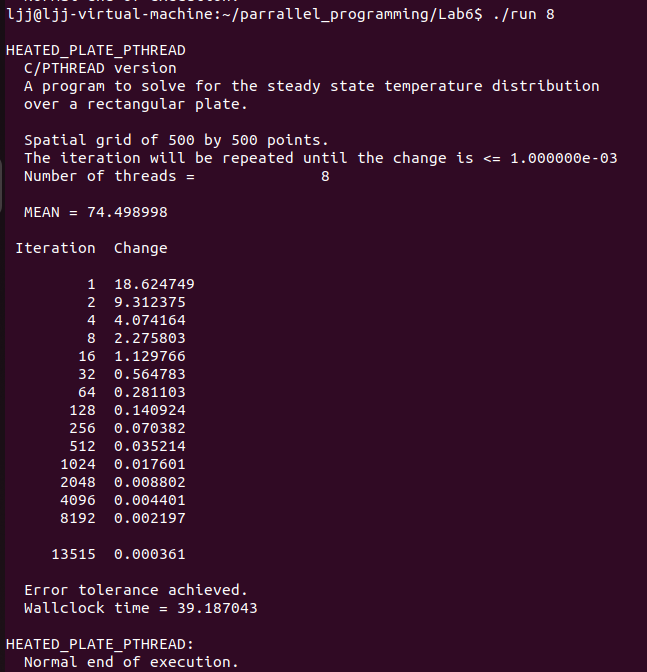


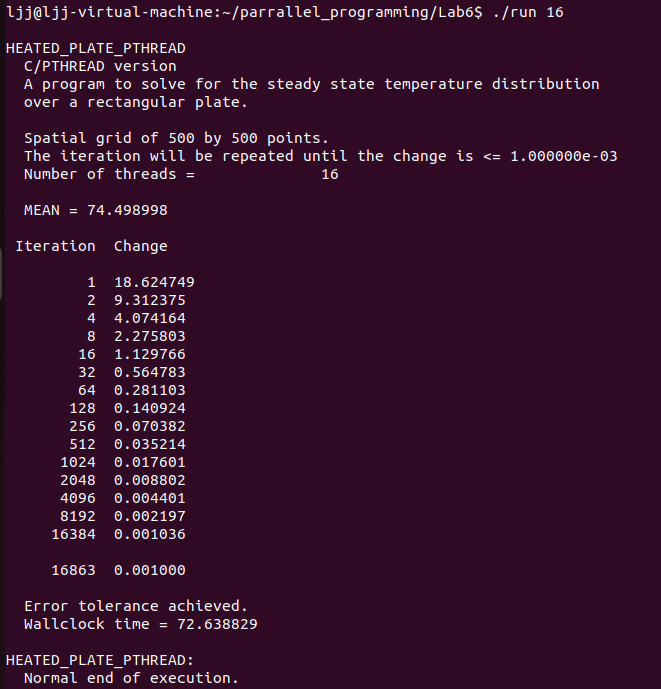






## pthread4





# 实验感想

**4.1问题**

**①程序运行过程中出现段错误**

**原因及解决办法：**

**②出现线程越多,运行时间越长的结果**

**原因及解决办法：**

**出现这种情况的原因是因为我在一开始的计时过程中使用了clock函数计时。**

**clock()函数的功能: 这个函数返回从“开启这个程序进程”到“程序中调用C++ clock()函数”时之间的CPU时钟计时单元（clock tick）数当程序单线程或者单核心机器运行时，这种时间的统计方法是正确的。但是如果要执行的代码多个线程并发执行时就会出问题，因为最终end-begin将会是多个核心总共执行的时钟嘀嗒数，因此造成时间偏大。**

**最后使用clock\_gettime()来计时,解决了这个问题。**