**实验报告**

姓名： 林炬乙 学号： 3180103721

课程名称： 计算机组成与系统结构 任课老师： 黄科杰

实验名称： 鲲鹏和矩阵乘法的并行计算 实验日期： 2021/6/4

# 1 实验目的和要求

（分点简要说明本次实验需要进行的工作和最终的目的）

1. 第一个实验室是在鲲鹏裸机服务器（BMS）上运行天气研究和预测（WRF）模型的并行计算。 实验室手册已在https://lab.huaweicloud.com/testdetail\_431上提供。 在本实验中，您将学习如何编译和运行 WRF，以及优化性能的方法。 请写下您对实验室的意见。

2. 第二个实验是运行矩阵乘法的并行计算（C=A\*B）。 所有矩阵都具有相同的 N\*N 维数。 请根据您在课堂和实验室中学到的知识编写代码。 请写一个非常简短的报告来显示并行计算的结果。 应该探索不同的矩阵维度。

# 2 实验原理

### 1. 测试使用time 命令

例如编译一个hello.c文件

#gcc hello.c -o hello

生成了hello可执行文件，此时统计该程序的运行时间便可以使用如下命令

#time ./hello   
在程序运行结束后便会显示出所需时间

real 0m2.913s

user 0m0.012s

sys 0m0.508s

可以大致认为程序执行的时间是 user 加上 sys ，其中 real 时间大于我们打印出来的程序执行时间是因为程序在执行时有系统阻塞，系统暂时去执行其他任务了。

### 2. 并行计算

我使用的是ubuntu18.04，配置起来比较简单，一般需要安装gcc的。

sudo apt-get install gcc

在编译之前我们需要加上-fopenmp，然后运行就可以执行我们带openmp的程序了。

gcc -fopenmp filename.cpp -o filename

schedule(dynamic)语句表示根据当前CPU负载动态分配线程，一般线程数为当前CPU的总线程数，每个CPU线程处理一个线程任务。

### 3. 准备benchmark

Python生成

f = open('log.txt','w')

for i in range(1,101,1):

    print(str(i),*end*= " ", *file*=f)

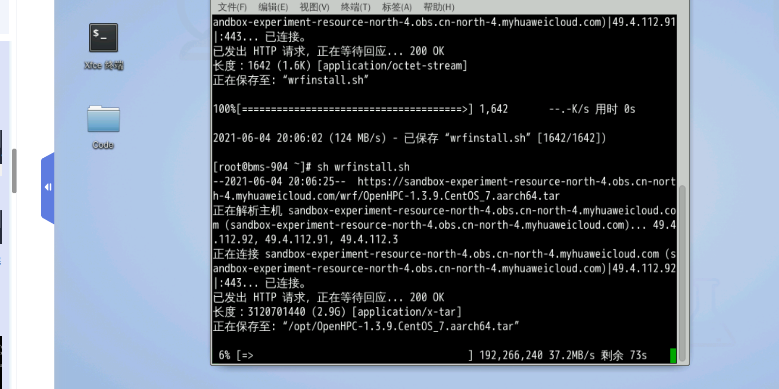
f.close()

//读入A,就像正常写入一样，把scanf换成fscanf 。fscanf(fp,"%d", &A[i][j]);

# 3 实验内容

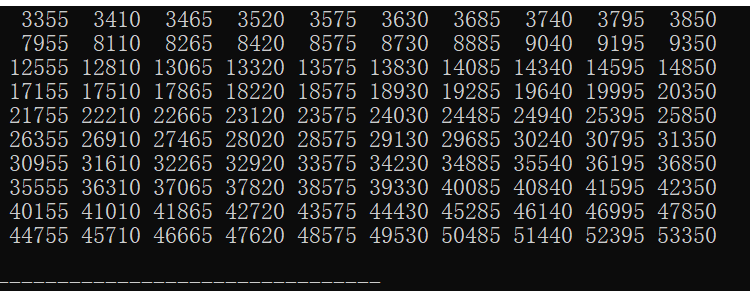
（分点阐述实验步骤）

连接华为云, 一开始只有私有地址没有公有地址, 就ssh 不上, 后来重启一次环境, 有公有地址了就可以了.



实验二， 首先我们写一个10\*10的矩阵

Devc可以运行



|  |  |  |
| --- | --- | --- |
| 行数=列数 | 并行 | 时间 |
| 100 | No | real 0m0.148s  user 0m0.004s  sys 0m0.004s |
| 100 | Yes | real 0m0.015s  user 0m0.026s  sys 0m0.000s  第二次real 0m0.152s  user 0m0.021s  sys 0m0.003s |
| 200 | Yes | support openmp  real 0m0.182s  user 0m0.166s  sys 0m0.012s |
| 200 | No |  |
| 400 | Yes | 4个线程  两个线程 |
| 400 | No |  |

## 遇到的错误:

1.File读取出错， 因为用了同一个fp，之前a读了读到末尾了，b就读取了0 ， 所以应该重新打开fp。

2. ssh连接服务器，发现连接失败，对应服务器的ip不能够ping通。

检查 一 : 云服务器的防火墙是否关闭

# 关闭防火墙

systemctl stop firewalld

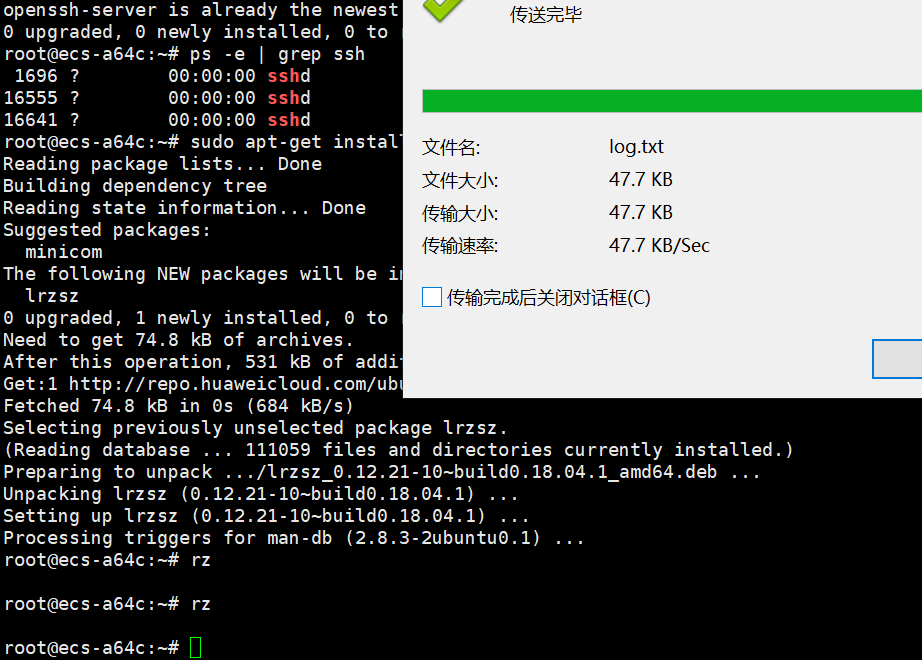
# 查看防火墙状态

systemctl status firewalld

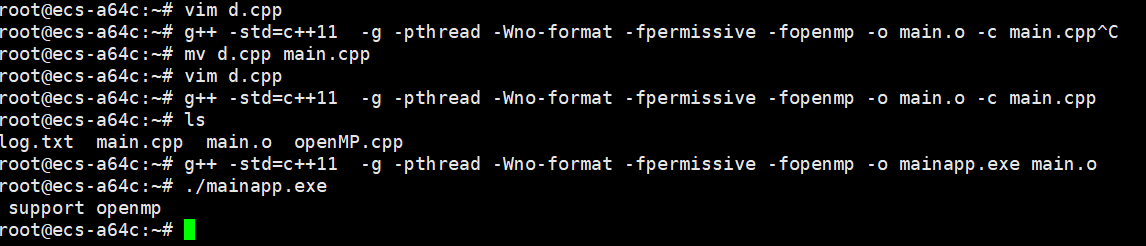
没有用

尝试2:

sudo apt-get install lrzsz

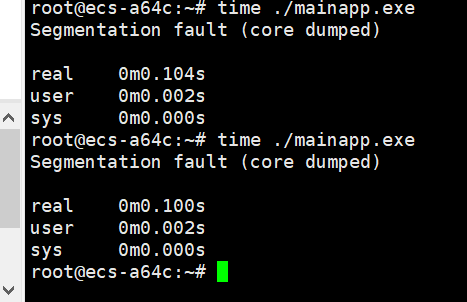


服务器自带g++的



即表明我们支持 OpenMP 了。

错误:



return value 3221225725

我室友说openmp是03年的,比较老, 矩阵太大会出错.

可能是int越界了

# 4 实验结果和分析

（使用图片和文字叙述实验结果，并对这些结果进行适当分析）

对华为实验的意见:

感觉可视化的地方不是很多, 很多机械的代码复制粘贴没有什么教育效果, 做了依然不知其所以然. 最好之后能可视化出来的成果, 了解这些代码实际产生了什么效果可以产生正反馈.对于并行计算的解释太少, 就最后一句话匆匆带过, 最好可以解释一下为什么200个并行线程? 少一点会如何? 多一些会如何? 核心少一点会如何? 多一些会如何?