

# Lab3

CS2305 Computer System Architecture

April 25, 2024

## 0 Requirements

- 最终需要提交一个压缩包，命名为 Lab3\_ 学号 \_ 姓名.zip，包含一个 pdf 文件和一个.c 文件。pdf 文件首页为封面，注明 “Lab3”，学号，姓名，专业等个人信息，第 2 页开始答题，无需抄题，但需注明题号。.c 文件依照题目要求给出你优化后的程序。
- 对于思考题，体现你的探索和思考，言之有理即可，不得抄袭，表述尽可能精简直接，允许中文作答。

# 1 Assignments

```
1 void matmul(int* L, int* R, int* result) {
2     for (int i = 0; i < L_ROW; i++) {
3         for (int j = 0; j < R_COL; j++) {
4             for (int k = 0; k < L_COL; k++) {
5                 result[i * R_COL + j] += L[i * L_COL + k] * R[k * R_COL + j];
6             }
7         }
8     }
9 }
```

上面是一段简单的矩阵乘代码，但是没有考虑到缓存结构等问题，效率较低。你的任务是，结合课程中关于 CPU 缓存结构与功能的知识，优化上述代码，使用循环重排、矩阵分块等方法，实现作业中留空的`optimized_matmul()`函数。

本次作业给出两个文件，分别是`data_gen.py`和`lab3.c`。`data_gen.py`中包含了生成随机  $L$ ,  $R$  矩阵数据作为输入，以及参考真值`true_value`。要使用`data_gen.py`，你需要安装`python`，以及`numpy`包。安装方法请自行查阅资料。

`data_gen.py`的格式如下：

```
1 python data_gen.py --L_row *** --L_col *** --R_row *** --R_col ***
```

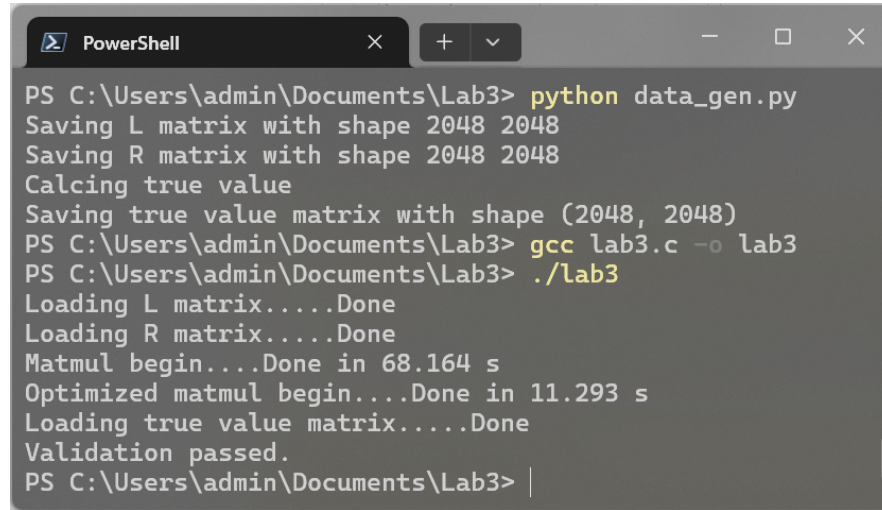
其中\*\*\*是对应矩阵维度的具体数值。所有维度的默认值均是 2048，即运行

```
1 python data_gen.py
```

会生成  $2048 \times 2048 \times 2048$  的  $L \times R$  数据，其中包含`L.dat`、`R.dat`、`true_value.dat`、`config.h`四个文件。从文件中读取数据的方法已在`lab3.c`中实现。

本次作业需要你提交一份 PDF 和你的程序。在 PDF 中给出你优化后的程序运行结果，随后结合代码说明你的程序使用了何种方法进行优化，并结合课程知识说明为什么这种方法可以带来程序性能的提升。此外，如果你实现的方法中包含可调整的参数，说明你使用的具体参数数值和确定该数值的方法。在本次作业中不允许使用 OpenMP 等多线程方法对程序加速或使用 Cuda 等非 CPU 的方法进行加速，但其他方法不做限制。

一次完整的测试流程如下图所示：



```
PS C:\Users\admin\Documents\Lab3> python data_gen.py
Saving L matrix with shape 2048 2048
Saving R matrix with shape 2048 2048
Calcing true value
Saving true value matrix with shape (2048, 2048)
PS C:\Users\admin\Documents\Lab3> gcc lab3.c -o lab3
PS C:\Users\admin\Documents\Lab3> ./lab3
Loading L matrix.....Done
Loading R matrix.....Done
Matmul begin....Done in 68.164 s
Optimized matmul begin....Done in 11.293 s
Loading true value matrix.....Done
Validation passed.
PS C:\Users\admin\Documents\Lab3> |
```

为了节省时间，你不需要为每次实验都重新生成数据。但你需要在提交作业之前执行几次上面的流程，确保优化后的算法不会出现计算错误。如果可以，你还可以探究关于 workload size 和优化后加速比之间的关系，并给出你的思考。