

LAB-5

The objective of this assignment is to study and compare the cache effects of Integer Matrix multiplication using the common naive algorithm and a Blocked matrix multiplication (Matrix multiplication with Cache blocking: a technique to improve cache performance by multiplying small blocks of matrix at a time)

This is an individual assignment.

The assignment has 2 parts:

1. Implement two algorithms to perform matrix multiplication of two integer square matrices, one is the naive algorithm and the other is a blocking algorithm. We have given a template C++ program. You are expected to fill in your implementation of the naive and blocked algorithm in part of the code marked by TODO comments. Please don't change any other code.
2. Use Perf tool to measure different hardware metrics related to the cache hierarchy (Number of references, Number of misses, miss rate, etc in the L1i, L1d and L2 caches) of the two algorithms for different sizes of input matrices (128x128, 256x256, 512x512). Observe the effects of changing the block size in the blocking matrix multiplication algorithm. What is the ideal block size? Compare the two algorithms based on the metrics.

Write a report summarizing the results and your observations. We have provided a template.cpp file that has to be used to fill in your implementation. There is also another file gen_input.cpp which can be used to generate random inputs to the matrix multiplication program by providing the size of the matrix as input. For example to generate an input file consisting of matrices of size 128, type the following commands:

```
$ g++ gen_input.cpp
```

```
$ ./a.out 128 input128
```

This will create a file input128 consisting of two 128x128 matrices to be used for matrix multiplication.

The syntax for executing the template.cpp file is

```
$ g++ template.cpp
```

```
$ ./a.out block_size inputfile outputfile
```

This will read the input matrices from inputfile and write its output to outputfile. If the block size is 0, then it uses native algorithm otherwise it will use blocked matrix multiplication using the block size given.

Submit a single zip file named as <rollno>.zip. For example if your roll number is CS19S026 the file should be named CS19S026.zip. The zip file should contain the completed C++ program named as <rollno>.cpp and your report.