Lab 9

Part 1

# of rows	average execution time for col_wise (micro seconds)	average execution time for row_wise (micro seconds)
10	72.017578	72.017578
100	9702.000000	9702.000000
1000	997002.000000	997002.000000
10000	18302494.000	18302494.000
50000	infinity loop	infinity loop

- What is spatial locality?

If you have data then it's likely your going to get data from a near by data adress.

- Based on your knowledge of spatial locality, which for loop should perform faster and why?

The outer loop, the inner loop is not going to take advantage of spatial locality

- According to the plot how does the execution time behave when we keep increasing the array size (which for loop is faster)?

The execution climbs and even does a infinity loop with the 10000 and 50000 array sizes, both loops where the same

- Is the answer you gave for 2 the same as the answer given for 3?

No, turns out for both the outer and inner loop I got the same execution times

- Why do you think the execution time behaves the way it does? Explain.

The time behaves that way due to the computer and the memory locations that need to be accessed

- Why do you think we executed each implementation 10 times? Explain.

To attempt to get a accurate avg for the output

Part 2

- Execute command lscpu in the terminal. What are the L1, L2 and L3 Cache sizes?

L1d cache: 768 KiB L1i cache: 768 KiB L2 cache: 12 MiB L3 cache: 192 MiB

- Execute command free -m in the terminal. What is the size of the RAM?

Mem: 515461 I available: 393134

- Does the RAM or cache sizes have an effect on the execution time?

Cache memory helds frequently used instruction/data which the processor may require next and it is faster access memory than RAM .since it is on the same chip as the processor.This reduce the need for frequent slower memory retrievals from main memory,which may otherwise keep the cpu waiting

Part 3

# of rows	col_wise execution (cache hits / cache misses)	row_wise execution (cache hits / cache misses)
10	00000/0	00000/1
100	000001/0	000001/0
1000	000010/0	000010/0
10000	000011/1	000011/1
50000	000100/1	000100/ 0

- What is the command you had to use to get the cache hits and misses?

/usr/bin/time -v

- What do the cache hits and misses indicate?

Data is not in cache memory so it pulls data from main memory and puts a block of data into cache layer.

- What is the relationship between cache hits, misses and spatial locality?

They take advantage a program's temporal and spatial locality