CS305 Assignment-9

Team:
Guttu Sai Abhishek 180050036
Hrithik Vikas Samala 180050038
Mulinti Shaik Wajid 180050063
Sai Phanindra Ramasahayam 180050084
Sanapathi Sumanth Balaji 180050091

Data type used in code: double

Observed values for matrix multiplication code(for mat-mult.cc):

S.no	size of matrix(n)	cache size(c)	block size(b)	associativity(a)	Total misses	Miss rate	Compulsory misses (fraction in total misses)	Capacity misses (fraction in total misses)	Conflict misses (fraction in total misses)
1	100	4k	16	1	826699	0.2051	15000 (0.0181)	510000 (0.6169)	301699 (0.3649)
2	100	4k	16	2	768558	0.1907	15000 (0.0195)	510000 (0.6636)	243558 (0.3169)
3	100	4k	16	4	619305	0.1537	15000 (0.0286)	510000 (0.9714)	94305 (0.1523)
4	100	16k	16	2	524936	0.1303	15000 (0.0286)	509936 (0.9714)	0 (0)
5	100	16k	32	2	267708	0.0664	7500 (0.0280)	254968 (0.9524)	5240 (0.0196)
6	100	16k	128	2	441842	0.1096	1876 (0.0042)	63920 (0.1447)	376046 (0.8511)
7	150	16k	16	2	1747582	0.1288	33750 (0.0193)	1709961(0.9785)	3871 (0.0022)

Observed values for matrix multiplication code(for mat-mult-opt.cc):

S.no	size of matrix(n)	cache size(c)	block size(b)	associativity(a)	Total misses	Miss rate	Compulsory misses (fraction in total misses)	Capacity misses (fraction in total misses)	Conflict misses (fraction in total misses)
1	100	4k	16	1	634709	0.1575	15000 (0.0236)	51000 (0.8035)	109709 (0.1728)
2	100	4k	16	2	530394	0.1316	15000 (0.0283)	510000 (0.9615)	5394 (0.0102)
3	100	4k	16	4	525000	0.1303	15000 (0.0286)	510000 (0.9714)	0(0)
4	100	16k	16	2	525000	0.1303	15000 (0.0286)	510000 (0.9714)	0(0)
5	100	16k	256	2	32864	0.0082	938 (0.0285)	31926 (0.9715)	0(0)
6	100	16k	2048	2	79253	0.0197	118 (0.0015)	4079 (0.0515)	75056 (0.9470)
7	150	16k	16	2	1747750	0.1288	33750 (0.0193)	1710000(0.9784)	4000 (0.0023)

Analysis:

1) As cache size increases miss rate decreases, conflict and capacity decreases, compulsory does not change.(2 to 4)
2) As block size increases miss rate decreases till some optimum value, after that miss rate increases with increase in block size, conflict misses increases, compulsory misses decreases(4 to 5, 5 to 6)
3) As associativity increases miss rate decreases with a high rate initially but as we go on increasing, this rate decreases. conflict misses decreases, capacity and compulsory misses does not change(1 to 2, 2 to 3)
4) As n changes there will not be much effect on miss rate but as number of memory access increase(as n increases), number of total misses and other types of misses also increases(4 to 7)

No.of lines in cache=cache size/block size

Explanations:

1)As cache size increases fixing block size, number of lines in cache increases so the number of lines in main memory which are mapped to a single cache line effectively decreases so conflict misses decrease. Compulsory misses do not change as block size is still the same. We are storing more data in cache and number of lines is high so total misses(miss rate) and capacity misses decreases

2)As block size increases for fixed cache size, number of lines in cache decreases so number of lines in main memory which are mapped to single cache line effectively increases so conflict misses increases and coming to miss rate we can see that it decreases from 4 to 5 because of spatial locality(getting more neighbourhood data at a single fetch) and increases from 5 to 6 is because of pollution of data(unwanted data). Compulsory misses decrease because of spatial locality.

In general if block size is increased capacity misses increase that is for a random set of access if block size is increased capacity misses increase because of increase in use of spatial locality but here we require it because of the continuous fetches from the main memory so capacity misses are decreasing here

3)As associativity increases conflict misses decreases (from definition of conflict misses that is if we take complete associativity we should get conflict misses as zero) we can see that from 1 to 2(for mat-mult-opt.cc) conflict misses decreases with very high rate(around 20 times) and from 2 to where associativity is 256(complete associativity) it is small decrease.

Conflict misses decreases as flexibility of replacement increases that is if associativity is 2 then for a conflict we can replace this from any of the 2 lines in set, capacity and compulsory misses does not change because of same configuration of cache as before which effect these such as block size and cache size (number of blocks) overall miss rate decreases.

4)As n increases fixing a,b,c same there is no effect on miss rate as it is a fraction of the number of misses in fetches, fetches and capacity, conflict, compulsory misses all increases with n(proportion remains same) as data fetched and number of fetches increase

Optimised vs Unoptimised:

-> We can observe that miss rate is less for mat-mult-opt.cc when compared to mat-mult.cc, this is because for mat-mult-opt, we are fetching D[j][k] (that is D[j][0],D[j][1]...)(D is transpose of C) instead of C[k][j] that is for one loop we are using elements from the same row (main memory stores matrix in row-major order) but in mat-mult we are fetching elements from different rows C[0][j] C[1][j] ….. So in mat-mult-opt we get less misses and for mat-mult we fetch one by one row which causes more misses.