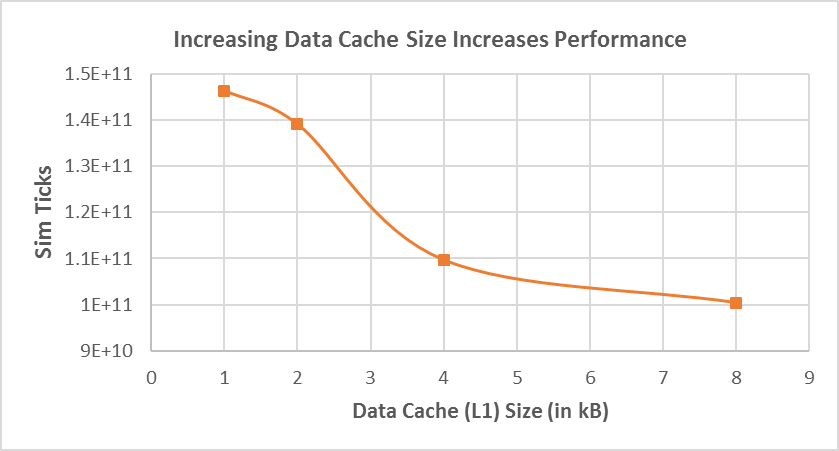
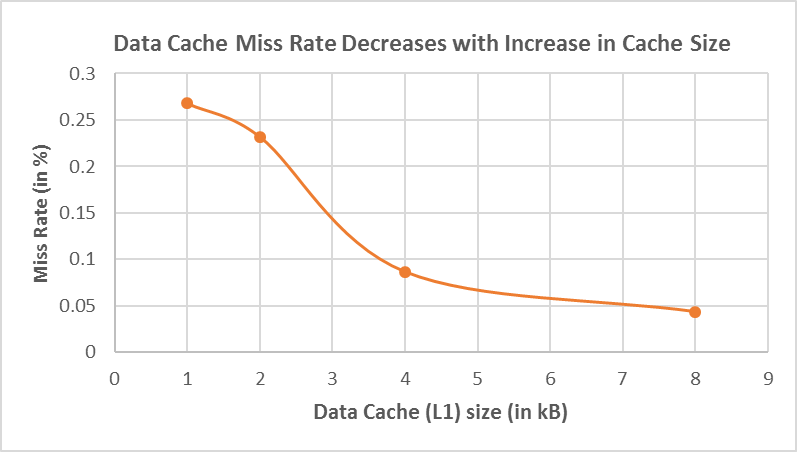
**Summer 2017: CSE 420: Computer Architecture 1**

**Project 2-Sample Solution**

**Problem 1)**

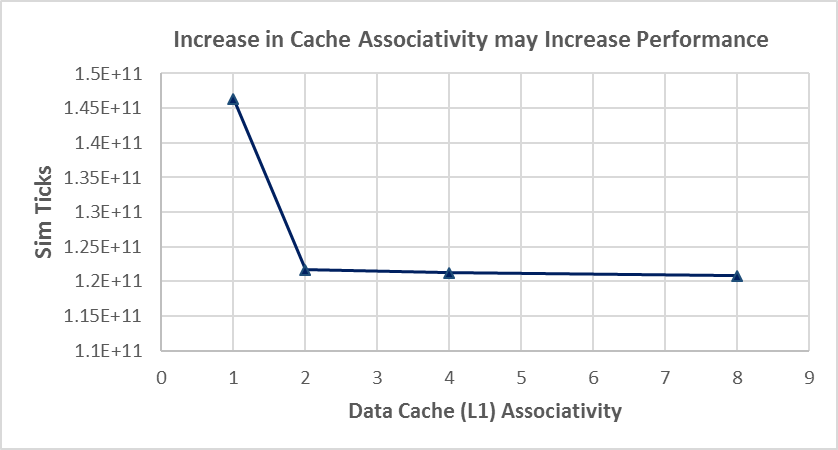
****

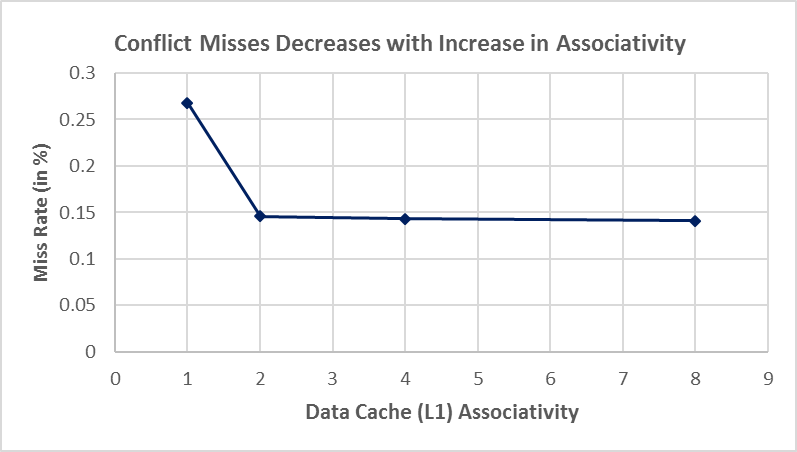
Although increasing cache size can result in potentially longer hit time and higher cost and power, it also allows the more data to fit in the cache i.e. more spatial and temporal locality to explore. So, miss rate goes down and performance increases.



**Problem 2)**

Increasing associativity obviously reduces conflict misses but at the cost of increased hit time and power consumption goes up. It may increase even miss penalty.

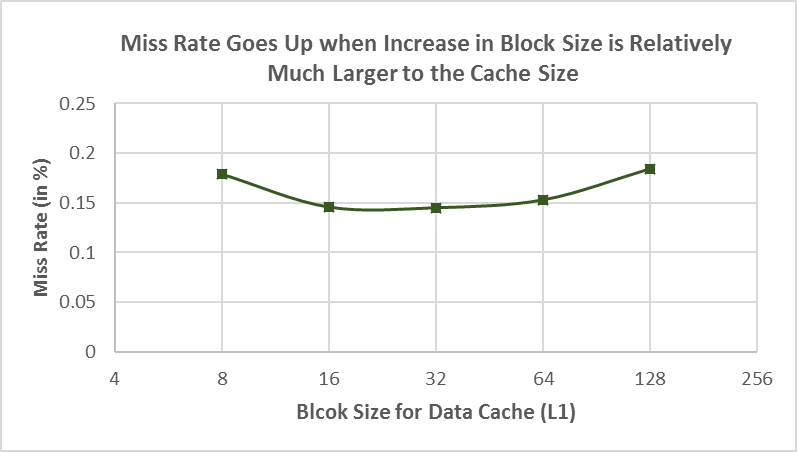




**Problem 3)**

Larger blocks help exploiting spatial locality, reducing miss rate. They reduce the compulsory miss but also increase the miss penalty. If the block size becomes significant fraction of the cache size (often in smaller caches), the number of blocks that can be held in the cache reduces and hence, the capacity or conflict misses increase with the increasing block size.

Choosing the right block size is always a trade-off that depends on the size of the cache vs. miss penalty and usually the block size is kept at 32 or 64 bytes.

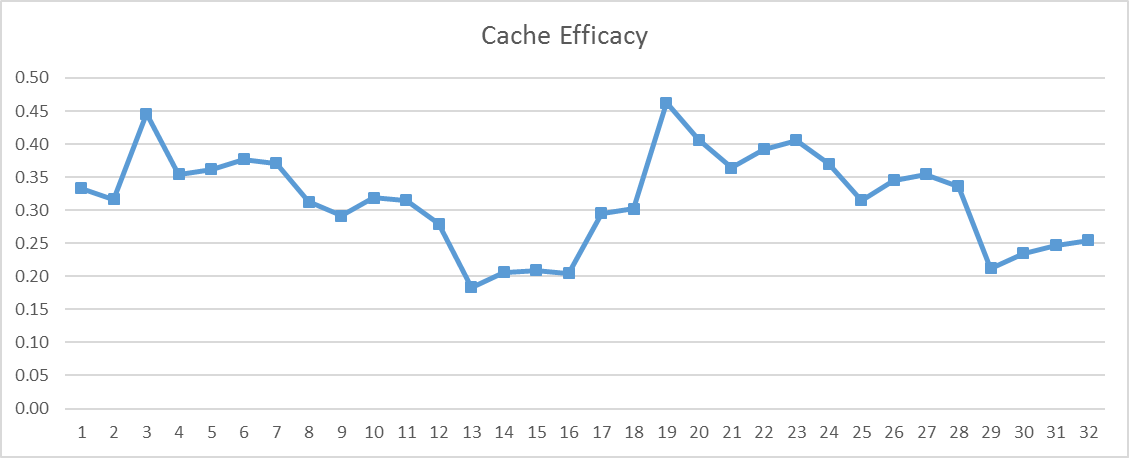


**Problem 4)**

**(a)**



**(b)**



**(c)**

Configuration 19 i.e. L1 cache of size 4 kB, associativity of 1 and cache line size of 32 is the optimal configuration.

**Problem 5)**

**(a)**

gem5/src/mem/cache

**(b)**

gem5/src/mem/cache/tags/lru.cc