

Problem 3

Even Flørenæs

Fall 2016

TDT4200 Parallel Computing
Department of Computer Science

Contents

1	Theory	3
1.1	Cache memory	3
1.2	Spatial and temporal locality	3
1.3	Cache coherence	3
1.4	False sharing	3

1 Theory

1.1 Cache memory

Fast memory which is closer to processor than the main memory. The purpose of cache memory is to store program instructions that are frequently referenced by software during operation. Fast access to these instructions increases the overall speed of the software program.

1.2 Spatial and temporal locality

Temporal locality refers to accessing a memory location defined somewhere in time. Spatial locality refers to accessing a memory location defined somewhere in the spatial domain.

Cache memories consider the spatial locality of memory accessing because if a particular memory location is referenced at one point, then it is likely that neighboring and generally nearby memory locations will be referenced in the near future.

1.3 Cache coherence

Cache coherence refers to the consistency of shared resource data that ends up stored in multiple local caches. In a multiprocessing system problems with inconsistent data may arise.

1.4 False sharing

False sharing is an action which may arise in a multithread system. If multiple threads with separate caches accesses different variables that belong to the same cache line, updates performed by one of these threads may force the other threads to copy from main memory even if no changes have happened to the data belonging to that particular thread. This action is performance-degrading, but ensures cache coherence.

1.5 OpenMP and Pthreads

Pthreads requires that the programmer explicitly specify the behavior of each thread. OpenMP, on the other hand, sometimes allows the programmer to

simply state that a block of code should be executed in parallel, and the precise determination of the tasks and which threads should execute them is left to the compiler and the run-time system. Pthread can be used with any C compiler, provided the system has a Pthreads library. OpenMP require compiler support for some operations. Pthreads is developed for usage at lower level programming, but OpenMP is developed to be used at higher level programming.