

NTIN066 - practicals 6

David Mareček

March 23, 2022

1. What happens in the tiled transposition for a cache-aware model with a weaker tall-cache assumption, i.e. when we do not have the cache size $M \geq 4B^2$, but only $M \geq B^2$?

Would it change the I/O complexity?

2. Prove that the “divide and conquer” algorithm for the cache-oblivious matrix transposition with naive recursion into 4 transpose and one swap operation lead to worse time complexity.

3. Analyze what happens in a recursive transposition algorithm if the size of the matrix is not a power of two.

Prove that sub-problems still work with almost square matrices.

4. Think about cache-aware and cache-oblivious strategies for static binary search trees.

5. Prove that for every cache size C and every $\epsilon > 0$, there exists a sequence of requests for which $T_{LRU} \geq (1 - \epsilon) \cdot C \cdot T_{OPT}$.