Weekplan: External memory II

Patrick Hagge Cording

References and Reading

[1] *"Cache oblivious B-trees"*, Michael A. Bender, Erik D. Demaine, Martin Farach-Colton.

[2] *"A locality-preserving cache-oblivious dynamic dictionary"*, Michael A. Bender, Ziyang Duan, John Iacono, Jing Wu.

Exercises

1 Binary search optimality. Prove that $O(\log_B N)$ IOs to do binary search is optimal in the comparison model ignoring lower order terms.

2 Hidden constants. The big-O notation hides constant factors in the number of IOs required for searching B-trees, but for practical purposes it is nice to know these.

2.1 Determine the worst case hidden constant in $O(\log_B N)$ IOs to do search in a B-tree in the external memory model.

2.2 Determine the worst case hidden constant in $O(\log_B N)$ IOs to do search in a B-tree in the cache oblivious model.

3 Batched searches. Suppose we want to search for a batch of at least M/B keys in a static search tree. When the query starts you know all the keys in the batch. Give a layout of the tree that will result in amortized $O(\log_B \frac{N}{M})$ IOs per key in the external memory model.

4 Matrix multiplication. You are given two matrices with dimensions $n \times m$ and $m \times p$.

4.1 Give an external memory algorithm that multiplies the two matrices using $O(\frac{nm}{B} + \frac{mp}{B} + \frac{m^2}{B} + \frac{nmp}{B\sqrt{M}})$ IOs.

4.2 Give a cache oblivious algorithm with the same bound on IOs.

5 Ordered-file maintenance. Prove the claim on slide 12.