Cubesort

From Wikipedia, the free encyclopedia

Cubesort is a parallel sorting algorithm that builds a self-balancing multi-dimensional array from the keys to be sorted. As the axes are of similar length the structure resembles a cube. After each key is inserted the cube can be rapidly converted to an array.[1]

A cubesort implementation written in C was published in 2014.[2]

Cubesort

Class	Sorting algorithm
Data structure	Array
Worst-case performance	$O(n \log n)$
Worst-case space complexity	$\Theta(n)$

Operation

Cubesort's algorithm uses a specialized binary search on each axis to find the location to insert an element. When an axis grows too large it is split. Locality of reference is optimal as only 4 binary searches are performed on small arrays for each insertion. By using many small dynamic arrays the high cost for insertion on single large arrays is avoided.

References

1. Cypher, Robert; Sanz, Jorge L.C (1992). "Cubesort: A parallel algorithm for sorting N data items with S-sorters".
2. "Cubesort".

External links

- Cubesort description and implementation in C (https://sites.google.com/site/binarysearchcube)
- Algorithms and Computation: 7th International Symposium, ISAAC '96, Osaka ... edited by Tetsuo Asano et al, pp 187-188, https://books.google.com/books?id=vilOl8JCpFUC&pg=PA188&lpg=PA188&hl=en&f=false (passing mention)