Exponential tree

From Wikipedia, the free encyclopedia

Exponential tree		
Type	Tree	
Invented	1995	
Invented by	Arne Andersson	
Time complexity in big O notation		
	Average	Worst case
Space	O(*n*)	O(*n*)
Search	O(min(√log *n*, log *n*/log *w*+ log log *n*, log *w* log log *n*))	O(min(√log *n*, log *n*/log *w*+ log log *n*, log *w* log log *n*))
Insert	O(min(√log *n*, log *n*/log *w*+ log log *n*, log *w* log log *n*))	O(min(√log *n*, log *n*/log *w*+ log log *n*, log *w* log log *n*))
Delete	O(min(√log *n*, log *n*/log *w*+ log log *n*, log *w* log log *n*))	O(min(√log *n*, log *n*/log *w*+ log log *n*, log *w* log log *n*))

An **exponential tree** is almost identical to a binary search tree, with the exception that the dimension of the tree is not the same at all levels. In a normal binary search tree, each node has a dimension (d) of 1, and has 2^d children. In an exponential tree, the dimension equals the depth of the node, with the root node having a d = 1. So the second level can hold two nodes, the third can hold eight nodes, the fourth 64 nodes, and so on.

Layout [edit]

"Exponential Tree" can also refer to a method of laying out the nodes of a tree structure in n (typically 2) dimensional space. Nodes are placed closer to a baseline than their parent node, by a factor equal to the number of child nodes of that parent node (or by some sort of weighting), and scaled according to how close they are. Thus, no matter how "deep" the tree may be, there is always room for more nodes, and the geometry of a subtree is unrelated to its position in the whole tree. The whole has a fractal structure.

In fact, this method of laying out a tree can be viewed as an application of the upper half-plane model of hyperbolic geometry, with isometries limited to translations only.

See also [edit]

- Faster deterministic sorting and searching in linear space (Original paper from '95)
- Laying out and Visualizing Large Trees Using a Hyperbolic Space
- Implementation and Performance Analysis of Exponential Tree Sorting

v · t · e	Tree data structures [hide]
Search trees (dynamic sets/associative arrays)	2–3 · 2–3–4 · AA · (a,b) · AVL · B · B+ · B* · B^x · (Optimal) Binary search · Dancing · HTree · Interval · Order statistic · (Left-leaning) Red-black · Scapegoat · Splay · T · Treap · UB · Weight-balanced
Heaps	Binary · Binomial · Fibonacci · Leftist · Pairing · Skew · Van Emde Boas
Tries	Hash · Radix · Suffix · Ternary search · X-fast · Y-fast
Spatial data partitioning trees	BK · BSP · Cartesian · Hilbert R · *k*-d (implicit *k*-d) · M · Metric · MVP · Octree · Priority R · Quad · R · R+ · R* · Segment · VP · X
Other trees	Cover · **Exponential** · Fenwick · Finger · Fusion · Hash calendar · iDistance · K-ary · Left-child right-sibling · Link/cut · Log-structured merge · Merkle · PQ · Range · SPQR · Top

This algorithms or data structures-related article is a stub. You can help Wikipedia by expanding it.

Categories: Exponentials | Trees (data structures) | Algorithms and data structures stubs | Computer science stubs