
Subject: Re: New containers - naming

Posted by [navi](#) on Sun, 03 Feb 2013 22:36:14 GMT

[View Forum Message](#) <> [Reply to Message](#)

wow. new container for NTL. awesome.

maybe an acronym prefix? like "aid" (Arbitrary insert delete)

e.g. aidArray, aidVector etc

also from description it sounds very similar to List.

Quote from wiki...Lists are typically implemented either as linked lists (either singly or doubly linked) or as arrays ...

... Each element in the list has an index. The first element commonly has index 0 or 1 (or some other predefined integer). Subsequent elements have indices that are 1 higher than the previous

It is possible to retrieve the element at a particular index.

It is possible to traverse the list in the order of increasing index.

It is possible to change the element at a particular index to a different value, without affecting any other elements.

It is possible to insert an element at a particular index. The indices of higher elements at that are increased by 1.

It is possible to remove an element at a particular index. The indices of higher elements at that are decreased by 1.

...Lists can be implemented as self-balancing binary search trees holding index-value pairs, providing equal-time access to any element (e.g. all residing in the fringe, and internal nodes storing the right-most child's index, used to guide the search), taking the time logarithmic in the list's size, but as long as it doesn't change much will provide the illusion of random access and enable swap, prefix and append operations in logarithmic time as well...

As the name implies, lists can be used to store a list of records. The items in a list can be sorted for the purpose of fast search (binary search)....

...lists are easier to realize than sets, a finite set in mathematical sense can be realized as a list with additional restrictions, that is, duplicate elements are disallowed and such that order is irrelevant. If the list is sorted, it speeds up determining if a given item is already in the set but in order to ensure the order, it requires more time to add new entry to the list. In efficient implementations, however, sets are implemented using self-balancing binary search trees or hash tables, rather than a list.