

GeeksforGeeks

A computer science portal for geeks

[Android App](#) [GeeksQuiz](#)

[Login/Register](#)

- [Home](#)
- [Algorithms](#)
- [DS](#)
- [GATE](#)
- [Interview Corner](#)
- [Q&A](#)
- [C](#)
- [C++](#)
- [Java](#)
- [Books](#)
- [Contribute](#)
- [Ask a Q](#)
- [About](#)

[Array](#)

[Bit Magic](#)

[C/C++](#)

[Articles](#)

[GFacts](#)

[Linked List](#)

[MCQ](#)

[Misc](#)

[Output](#)

[String](#)

[Tree](#)

[Graph](#)

Linked List vs Array

Difficulty Level: Rookie

Both Arrays and [Linked List](#) can be used to store linear data of similar types, but they both have some advantages and disadvantages over each other.

Following are the points in favour of Linked Lists.

(1) The size of the arrays is fixed: So we must know the upper limit on the number of elements in advance. Also, generally, the allocated memory is equal to the upper limit irrespective of the usage, and in practical uses, upper limit is rarely reached.

(2) Inserting a new element in an array of elements is expensive, because room has to be created for the new elements and to create room existing elements have to be shifted.

For example, suppose we maintain a sorted list of IDs in an array `id[]`.

`id[] = [1000, 1010, 1050, 2000, 2040,]`.

And if we want to insert a new ID 1005, then to maintain the sorted order, we have to move all the elements after 1000 (excluding 1000).

Deletion is also expensive with arrays unless some special techniques are used. For example, to delete 1010 in `id[]`, everything after 1010 has to be moved.

So Linked list provides following two advantages over arrays

- 1) Dynamic size
- 2) Ease of insertion/deletion

Linked lists have following drawbacks:

- 1) Random access is not allowed. We have to access elements sequentially starting from the first node. So we cannot do binary search with linked lists.
- 2) Extra memory space for a pointer is required with each element of the list.
- 3) Arrays have better cache locality that can make a pretty big difference in performance.

Please also see [this](#) thread.

References:

<http://cslibrary.stanford.edu/103/LinkedListBasics.pdf>

Please write comments if you find anything incorrect, or you want to share more information about the topic discussed above.

Related Topics:

- [Divide and Conquer | Set 7 \(The Skyline Problem\)](#)
- [Divide and Conquer | Set 6 \(Tiling Problem\)](#)
- [Count number of islands where every island is row-wise and column-wise separated](#)
- [Rearrange an array such that 'arr\[j\]' becomes 'i' if 'arr\[i\]' is 'j'](#)
- [Find position of an element in a sorted array of infinite numbers](#)
- [Find a common element in all rows of a given row-wise sorted matrix](#)
- [Can QuickSort be implemented in O\(nLogn\) worst case time complexity?](#)
- [Fill two instances of all numbers from 1 to n in a specific way](#)

Like < 18

Tweet < 4

 < 2

Writing code in comment? Please use ideone.com and share the link here.

-
-
-
- - [Interview Experiences](#)
 - [Advanced Data Structures](#)
 - [Dynamic Programming](#)
 - [Greedy Algorithms](#)
 - [Backtracking](#)
 - [Pattern Searching](#)
 - [Divide & Conquer](#)
 - [Mathematical Algorithms](#)
 - [Recursion](#)
 - [Geometric Algorithms](#)
-

• Popular Posts

- [All permutations of a given string](#)
- [Memory Layout of C Programs](#)
- [Understanding “extern” keyword in C](#)
- [Median of two sorted arrays](#)
- [Tree traversal without recursion and without stack!](#)
- [Structure Member Alignment, Padding and Data Packing](#)
- [Intersection point of two Linked Lists](#)
- [Lowest Common Ancestor in a BST.](#)
- [Check if a binary tree is BST or not](#)
- [Sorted Linked List to Balanced BST](#)

Follow @GeeksforGeeks



[Subscribe](#)

• Recent Comments

- [lebron](#)

since the array size is 5, it takes constant...

[K'th Smallest/Largest Element in Unsorted Array | Set 3 \(Worst Case Linear Time\)](#) · [2 hours ago](#)

- [lebron](#)

merge sort

[K'th Smallest/Largest Element in Unsorted Array | Set 3 \(Worst Case Linear Time\)](#) · [2 hours ago](#)

- [Shubham Sharma](#)

You saved my time :)

[Searching for Patterns | Set 2 \(KMP Algorithm\)](#) · [3 hours ago](#)

- Prakhar

Why so many LOCs, if I'm not wrong (please...

[Largest Sum Contiguous Subarray](#) · [3 hours ago](#)

- [Aayush Gupta](#)

For R4 Q3, Another solution would be to use a...

[Amazon Interview Experience | Set 168](#) · [4 hours ago](#)

- [EigenHarsha](#)

For Power Of 2, We Simply Doing.. var1 =

[Practo Interview Experience | Set 2 \(Off-Campus\)](#) · [4 hours ago](#)

•

@geeksforgeeks, [Some rights reserved](#) ____ [Contact Us!](#)

Powered by [WordPress](#) & [MooTools](#), customized by geeksforgeeks team