Linked List vs Array

Last Updated : 17 Feb, 2025

Array: Arrays store elements in contiguous memory locations, resulting in easily calculable addresses for the elements stored and this allows faster access to an element at a specific index.

Data storage scheme of an array

Linked List: Linked lists are less rigid in their storage structure and elements are usually not stored in contiguous locations, hence they need to be stored with additional tags giving a reference to the next element.

Linked-List representation

Advantages of Linked List over arrays :

- **Efficient insertion and deletion**: Linked lists allow insertion and deletion in the middle in O(1) time, if we have a pointer to the target position, as only a few pointer changes are needed. In contrast, arrays require O(n) time for insertion or deletion in the middle due to element shifting.
- **Implementation of Queue and Deque** : Simple array implementation is not efficient at all. We must use circular array to efficiently implement which is complex. But with linked list, it is easy and straightforward. That is why most of the language libraries use Linked List internally to implement these data structures.
- **Space Efficient in Some Cases :** Linked List might turn out to be more space efficient compare to arrays in cases where we cannot guess the number of elements in advance. In case of arrays, the whole memory for items is allocated together. Even with dynamic sized arrays like vector in C++ or list in Python or ArrayList in Java. the internal working involves de-allocation of whole memory and allocation of a bigger chunk when insertions happen beyond the current capacity.
- **Circular List with Deletion/Addition :** Circular Linked Lists are useful to implement CPU round robin scheduling or similar requirements in the real world because of the quick deletion/insertion in a circular manner.

Advantages of Arrays over Linked List :

- **Random Access**. : We can access ith item in O(1) time (only some basic arithmetic required using base address). In case of linked lists, it is O(n) operation due to sequential access**.**
- **Cache Friendliness** : Array items (Or item references) are stored at contiguous locations which makes array cache friendly (Please refer Spatial locality of reference for more details)
- **Easy to use :** Arrays are relatively very easy to use and are available as core of programming languages
- **Less Overhead :** Unlike linked list, we do not have any extra references / pointers to be stored with every item.

Comment | More info | Advertise with us

Next Article

Types of Linked List

Similar Reads

Linked List Data Structure

A linked list is a fundamental data structure in computer science. It mainly allows efficient insertion and deletion operations compared to arrays. Like arrays, it is also used to implement other data structures like stack, queue and deque. Hereâ€™s the comparison of Linked List vs Arrays Linked List:

3 min read

Basic Terminologies of Linked List

Linked List is a linear data structure, in which elements are not stored at a contiguous location, rather they are linked using pointers. Linked List forms a series of connected nodes, where each node stores the data and the address of the next node. Node Structure: A node in a linked list typically

3 min read

Introduction to Linked List - Data Structure and Algorithm Tutorials

Linked List is basically chains of nodes where each node contains information such as data and a pointer to the next node in the chain. It is a popular data structure with a wide range of real-world applications. Unlike Arrays, Linked List elements are not stored at a contiguous location. In the lin

10 min read

Applications, Advantages and Disadvantages of Linked List

A Linked List is a linear data structure that is used to store a collection of data with the help of nodes. Please remember the following points before moving forward. The consecutive elements are connected by pointers / references.The last node of the linked list points to null.The entry point of a

4 min read

Linked List vs Array

Array: Arrays store elements in contiguous memory locations, resulting in easily calculable addresses for the elements stored and this allows faster access to an element at a specific index. Linked List: Linked lists are less rigid in their storage structure and elements are usually not stored in co

2 min read

Types of Linked List

Basic Operations on Linked List

Top 50 Problems on Linked List Data Structure asked in SDE Interviews

A Linked List is a linear data structure that looks like a chain of nodes, where each node is a different element. Unlike Arrays, Linked List elements are not stored at a contiguous location. Here is the collection of the Top 50 list of frequently asked interview questions on Linked Lists. Problems

3 min read
