

[Related Articles](#)[Save for later](#)

# Difference Between Vector and List

Difficulty Level : Easy ● Last Updated : 28 May, 2020

**Vector:** Vector is a type of dynamic array which has the ability to resize automatically after insertion or deletion of elements. The elements in vector are placed in contiguous storage so that they can be accessed and traversed using iterators. Element is inserted at the end of the vector.

## Example:

```
vector v;  
v.insert(5);  
v.delete();
```

**List:** List is a double linked sequence that supports both forward and backward traversal. The time taken in the insertion and deletion in the beginning, end and middle is constant. It has the non-contiguous memory and there is no pre-allocated memory.

## Example:

```
list l;  
l.insert_begin(5);  
l.delete_end();
```



We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).

**Got It !**



Below is a table of differences between Vector and List:

Vector	List
It has contiguous memory.	While it has non-contiguous memory.
It is synchronized.	While it is not synchronized.
Vector may have a default size.	List does not have default size.
In vector, each element only requires the space for itself only.	In list, each element requires extra space for the node which holds the element, including pointers to the next and previous elements in the list.
Insertion at the end requires constant time but insertion elsewhere is costly.	Insertion is cheap no matter where in the list it occurs.
Vector is thread safe.	List is not thread safe.
Deletion at the end of the vector needs constant time but for the rest it is $O(n)$ .	Deletion is cheap no matter where in the list it occurs.

We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).

**Got It !**

Vector	List
Iterators become invalid if elements are added to or removed from the vector.	Iterators are valid if elements are added to or removed from the list.



Like 0

Previous

**What is a Motherboard?**

Next

**Difference between  
Cryptography and  
Cryptology**

RECOMMENDED ARTICLES

Page : 1 2 3

We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).

**Got It !**

**01** Difference between `std::swap` and `std::vector::swap`

28, Sep 18

**02** Difference between Distance vector routing and Link State routing

30, Apr 18

**03** Difference between `std::remove` and `vector::erase` for vectors

18, Jul 17

**04** Difference between `std::set` vs `std::vector` in C++ STL

17, Nov 20

**05** Difference between Singly linked list and Doubly linked list

07, Jan 19

**06** Difference between forward list and list in C++

08, Jul 20

**07** Difference between List and Array in Python

22, Jun 20

**08** Difference between List comprehension and Lambda in Python

05, Jun 20

We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).

**Got It !**

## Article Contributed By :



**pp\_pankaj**

@pp\_pankaj

## Vote for difficulty

Current difficulty : [Easy](#)

Easy

Normal

Medium

Hard

Expert

Article Tags : [Difference Between](#), [GBlog](#)

Improve Article

Report Issue

Courses

Company-wise

Topic-wise

How to begin?

write an article

Write Interview Experience

Internships

Videos

@geeksforgeeks , Some rights reserved

We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).

**Got It !**