

Difference Between Stack and Queue

Having discussed the data structures, their types, stack data structure, and queue data structure, now we will see the difference between Stack and Queue. Though both stack and queue belong to the linear data structures, differences exist based on the operations and features each offers to the programmer or user.

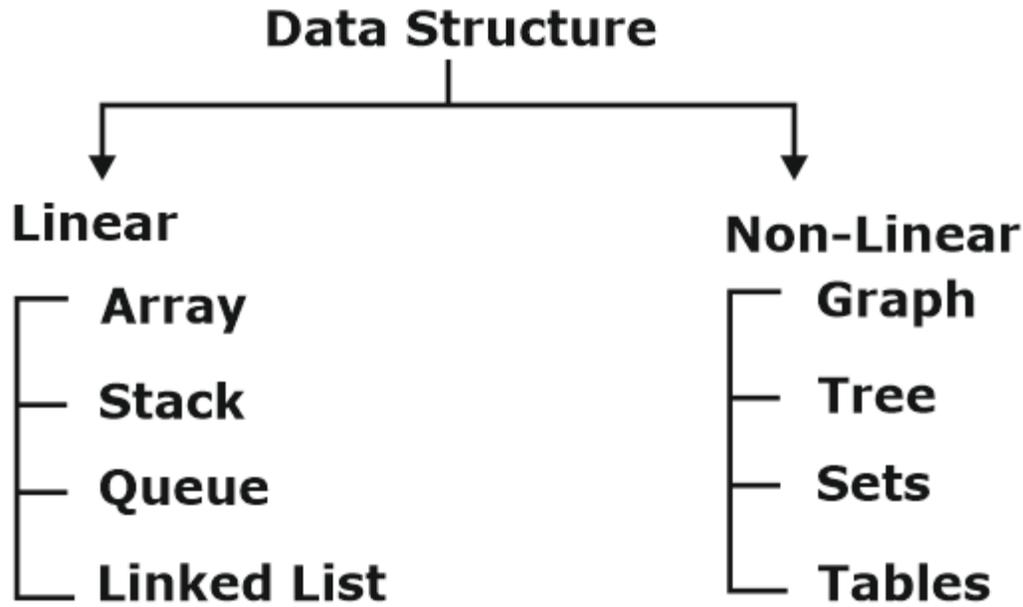
Key Differences Between Stack and Queue

Stack	Queue
Elements are inserted or deleted from the same end, called the stack's top.	Elements are inserted and deleted respectively from the rear and front end of the queue.
Stack follows LIFO	Queue follows FIFO
Stack uses push and pop operations to insert and delete data, respectively.	Queue uses 'en queue' and 'de queue' operations to insert and delete data.
A stack has no further division.	A queue is divided into the circular queue, priority queue, and double-ended queue.
Implementation is simple.	Implementation is complex.
Used in expression evaluation, undo, and redo operations.	Used in assigning processes, ready queue, waiting for a queue, etc.

What is a Data Structure?

Data may be organized in many different ways. The logical or mathematical organization of data is called the data structure. Programming languages use data structures to store, manipulate, access, and organize data. Algorithms are implemented using these data structures.

The data structures are broadly classified into two categories based on their operations. The two categories are a Linear data structure and a Non-linear data structure. We have linear arrangements of data in the linear data structure, while in the non-linear data structure, the data is stored in the form of trees, tables, etc. Despite the difference between Stack and Queue, both stack and queue belong to the linear data structures.



What is a Stack Data Structure?

A stack is an abstract data type (ADT). A real-world stack allows operation at one end only. At any given time, we can only access the top element of a stack. This feature makes it a LIFO data structure. LIFO stands for Last In First Out. The operational difference between stack and queue involves LIFO and FIFO (First In, First Out); we will learn about FIFO in the queue data structure. The basic operations of the stack data structure are:

1. push(): Insert an element into the stack.
2. pop(): Removing an element from the stack.
3. peek(): Get the top element of the stack without removing it.
4. isFull() : Checks if the stack is full.
5. isEmpty(): Checks if the stack is empty.

What is a Queue Data Structure?

A queue is an abstract data structure similar to a stack. Unlike stacks, a queue is open at both ends. One end is always used to insert data, and the other is to remove data from the queue. Queue follows the FIFO methodology. The end at which insertions take place is called the rear end of the queue and the end at which deletion takes place is called the front of the queue. The basic operations of queue data structures are:

- enqueue() : To insert an item into the queue.
- dequeue() : To remove an element from the queue.

- peek(): To get the element from the front of the queue without removing it.
- isFull() : Checks if the queue is full.
- isEmpty() : Checks if the queue is empty.

