Intro to data structure & algorithms

by Mariana Makram / ITI-sohag

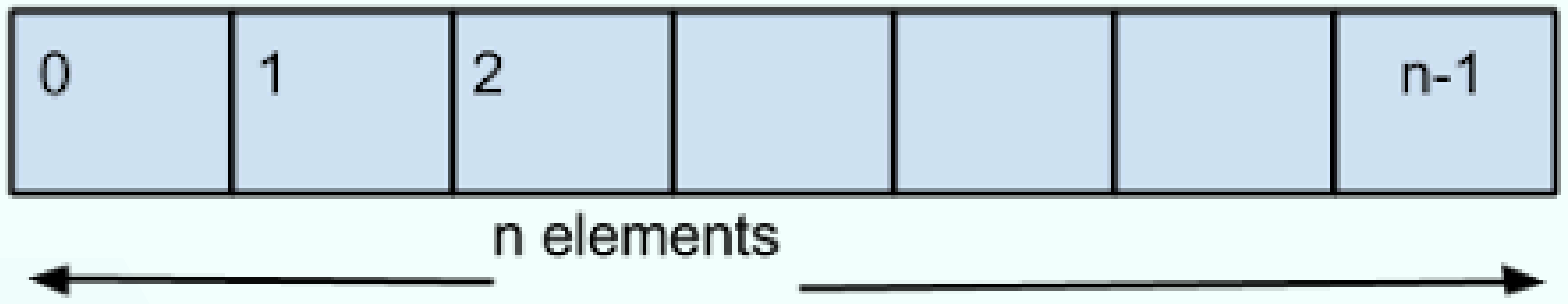
What is Data Structure

Data Structures are the programmatic way of organizing, managing, and storing data so that data can be used efficiently.

With the help of Data Structures, we can easily traverse, add and manipulate the data items.

Array as Data Structure

Arrays are defined as the collection of similar types of data items stored at contiguous memory locations. It is one of the simplest data structures where each data element can be randomly accessed by using its index number.



Array as Data Structure

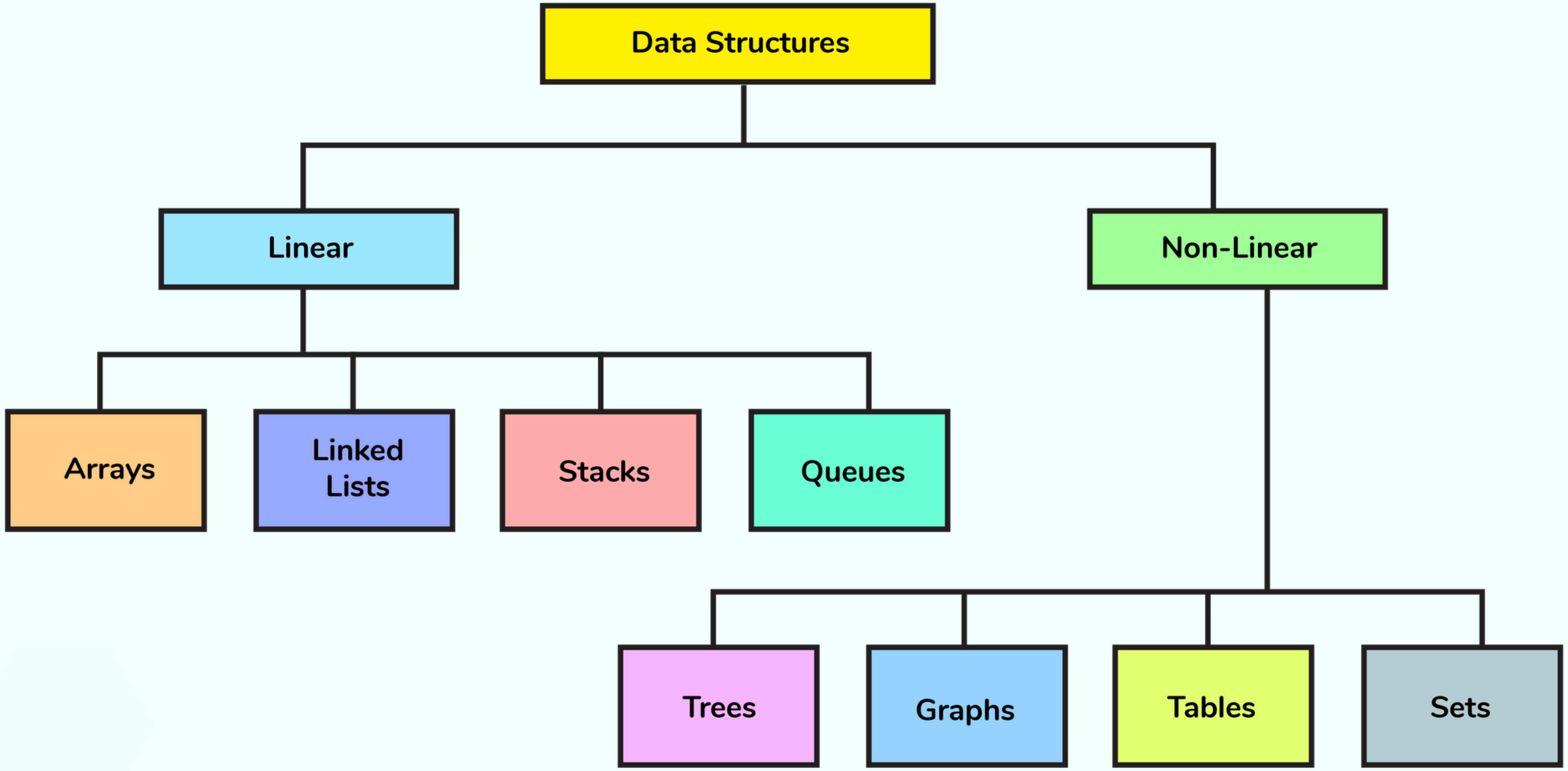
Advantages

- Any element in the array can be directly accessed by using the index.
- Traversing an array is a very simple process; we just need to increment the base address of the array in order to visit each element one by one

Disadvantages

- In array, there is static memory allocation that is size of an array cannot be altered.
- There will be wastage of memory if we store less number of elements than the declared size.

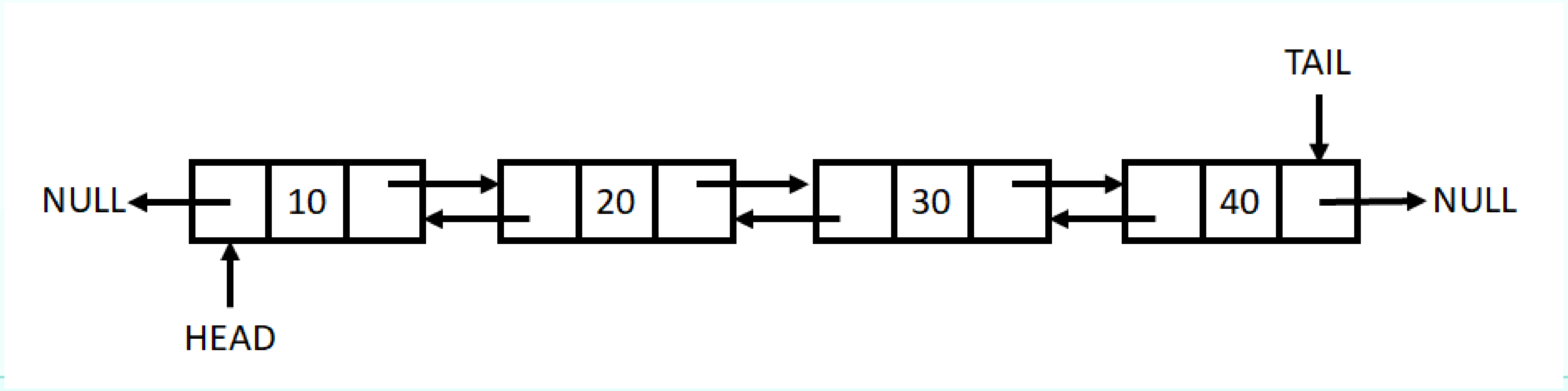
Classification of Data Structures



Linked list

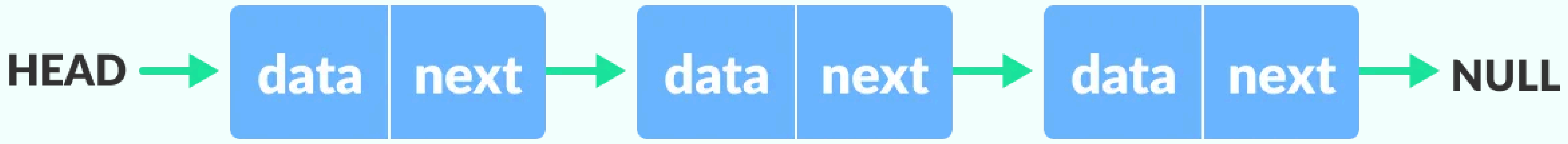
- Linked list is a linear data structure that includes a series of connected nodes. Linked list can be defined as the nodes that are randomly stored in the memory.

- In linked list, size is no longer a problem since we do not need to define its size at the time of declaration.

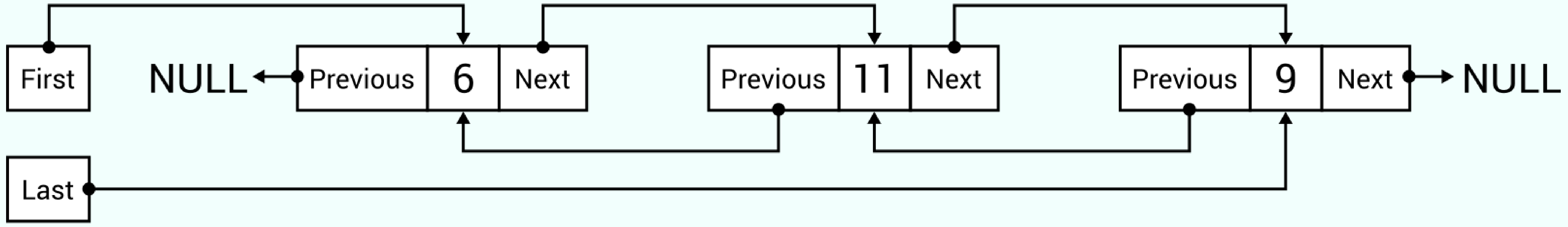


Types of Linked list

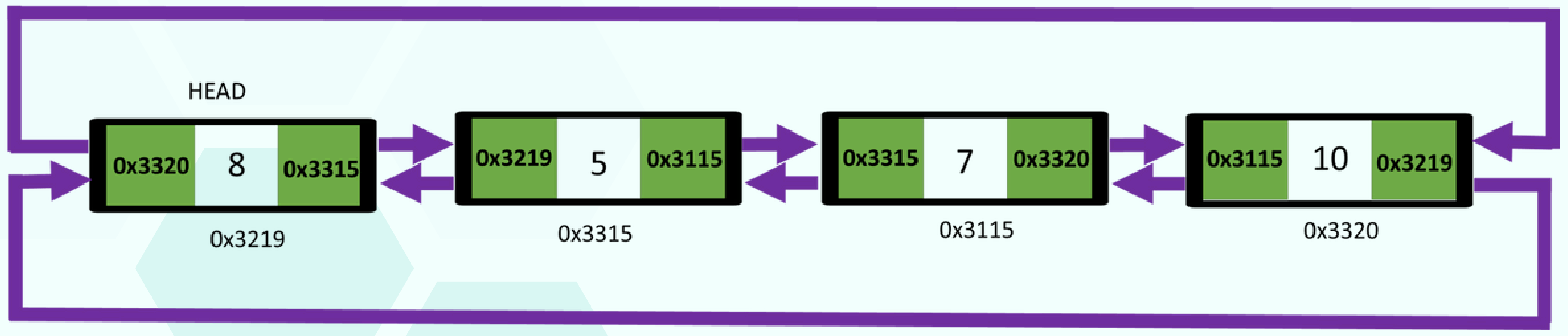
- Singly-linked list

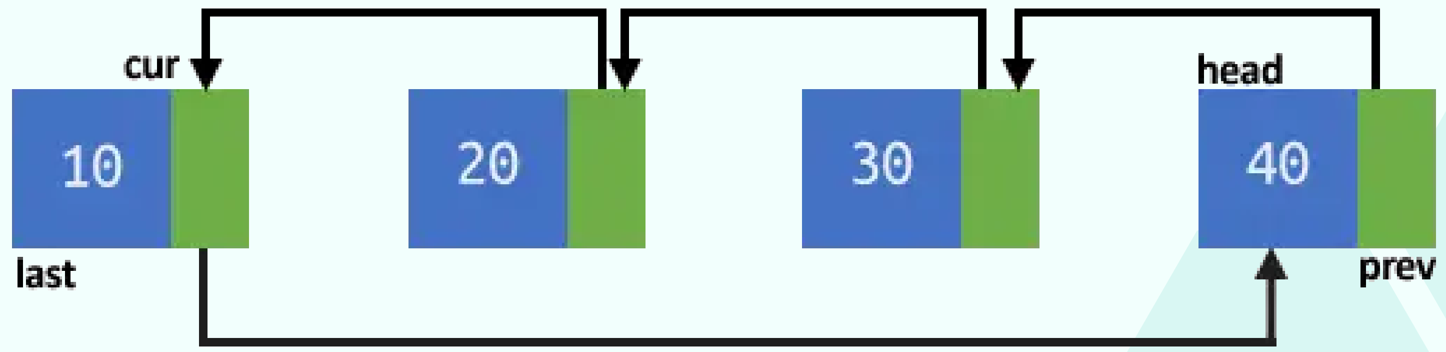


- Doubly linked list



- Circular linked list





Linked list

Advantages

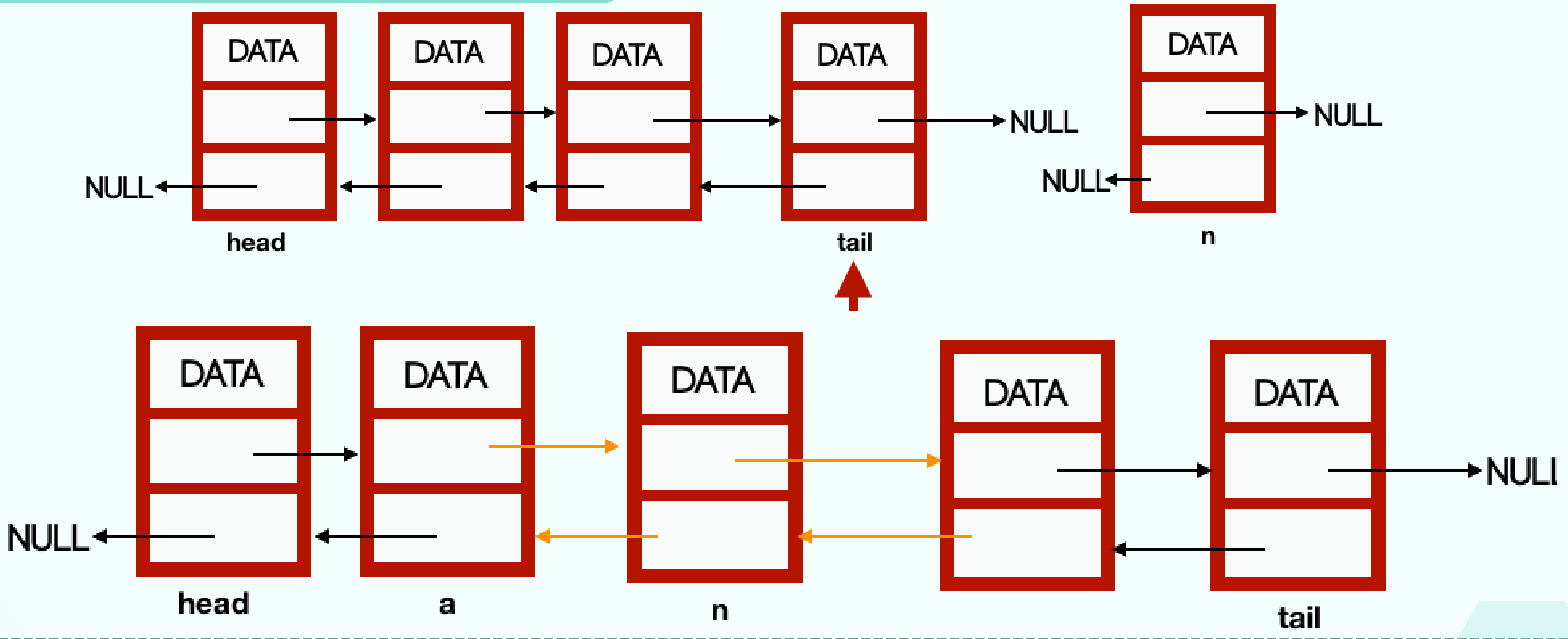
- Dynamic data structure - Linked list does not have a fixed size. can grow or shrink according to the requirements.
- Insertion and deletion -we just have to update the address of the pointer of the node.

Disadvantages

- Memory usage -Each node occupies two types of variables, i.e., one is a simple variable, and another one is the pointer.
- If we have to access an element we need to traverse all the nodes before it.

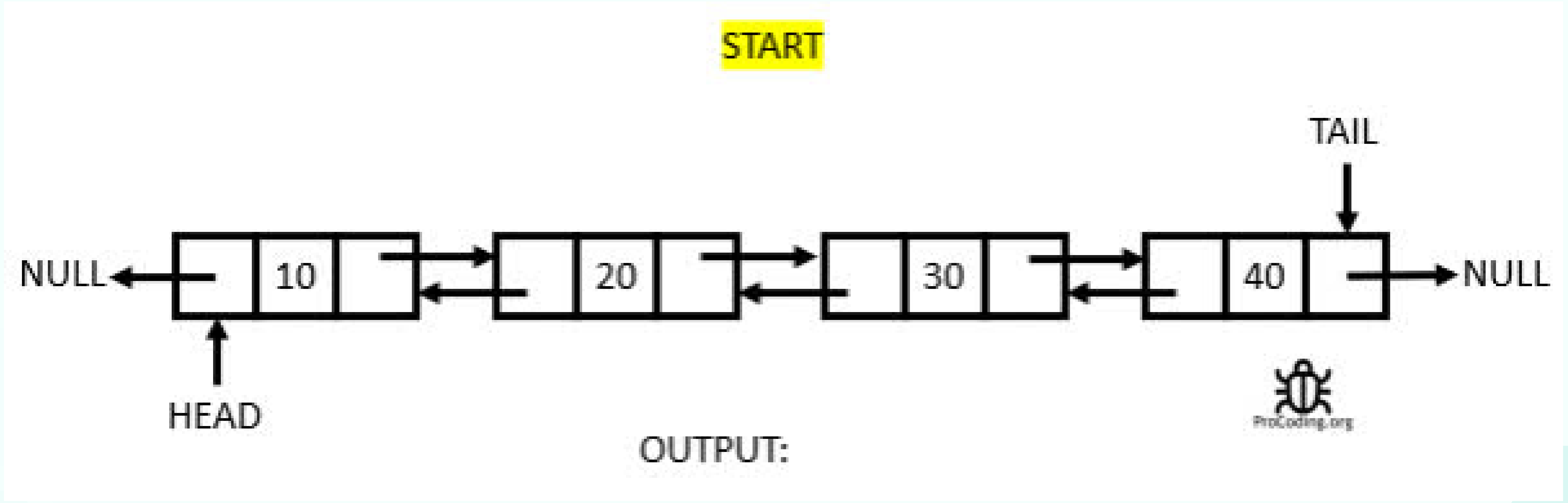
Operations on Linked list

Add new node



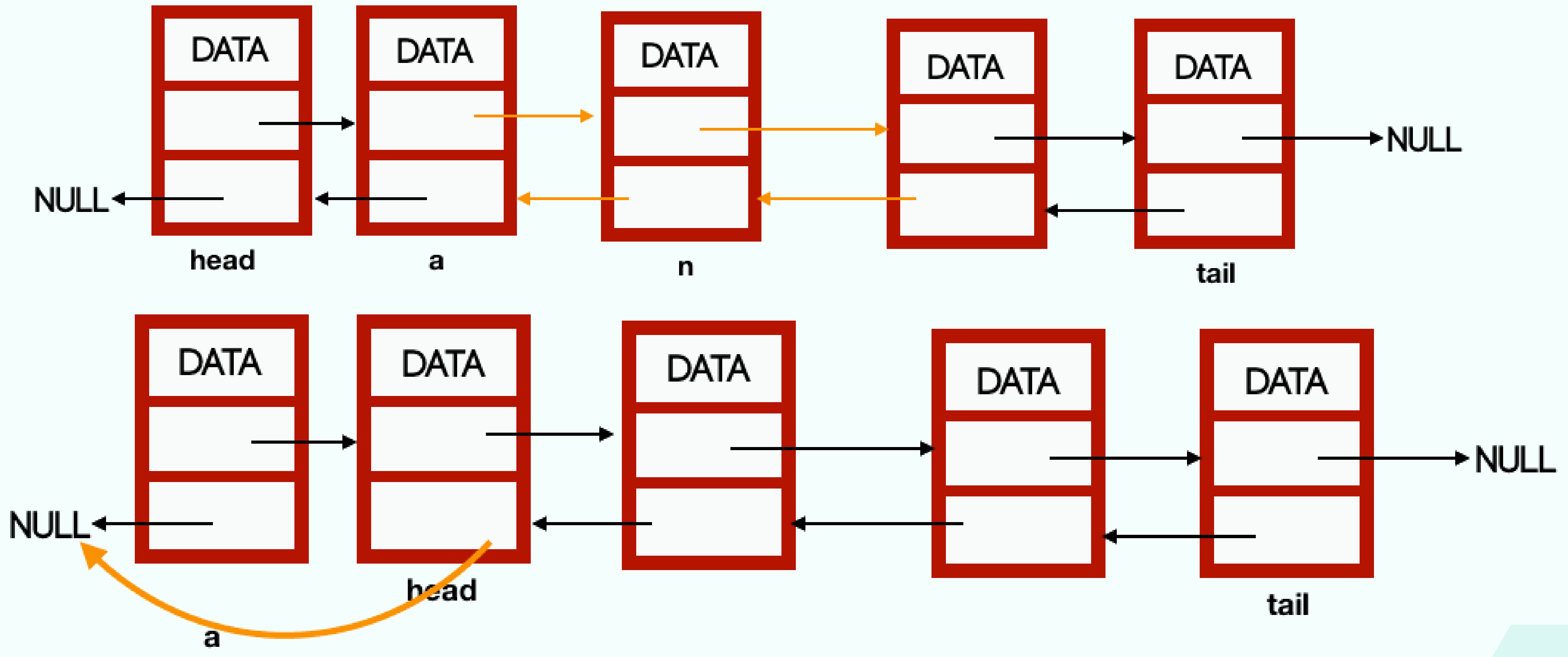
Operations on Linked list

Display linked list



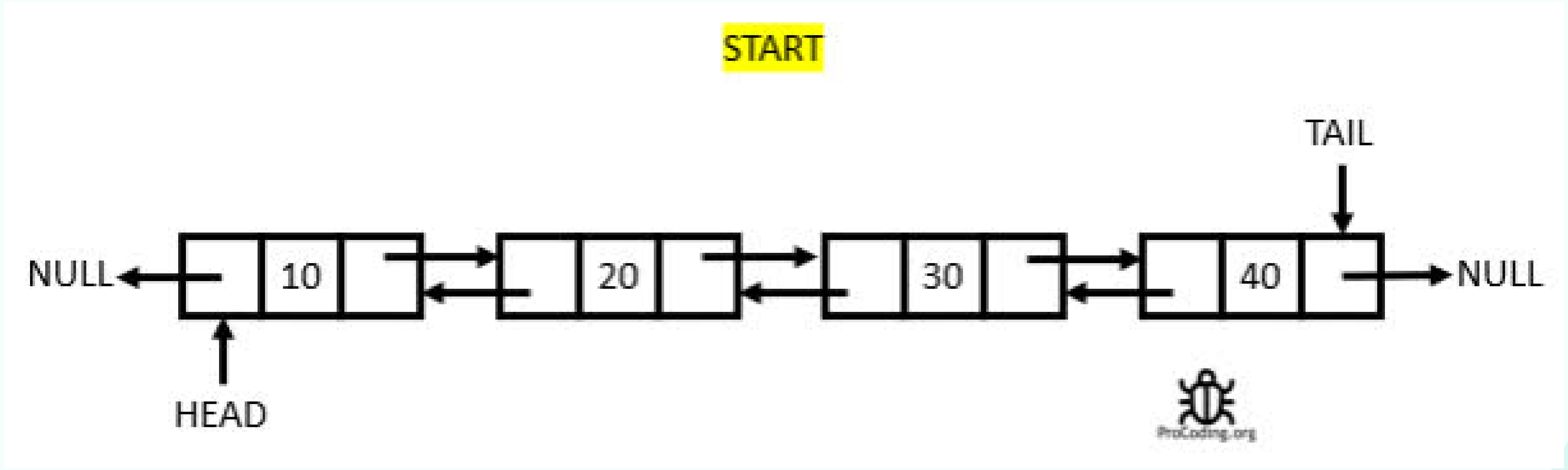
Operations on Linked list

Delete Node (Head)



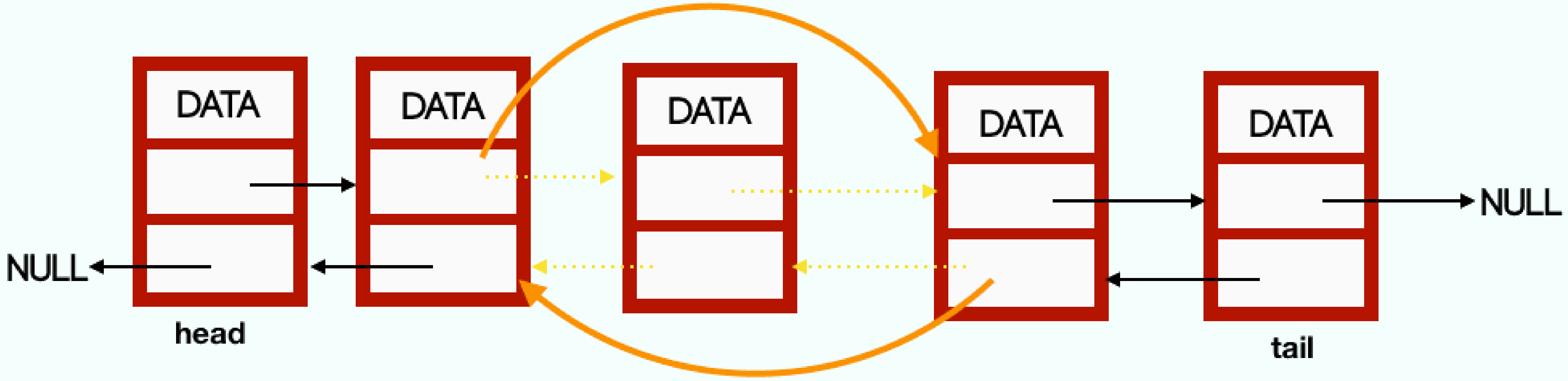
Operations on Linked list

Delete Node (Tail)



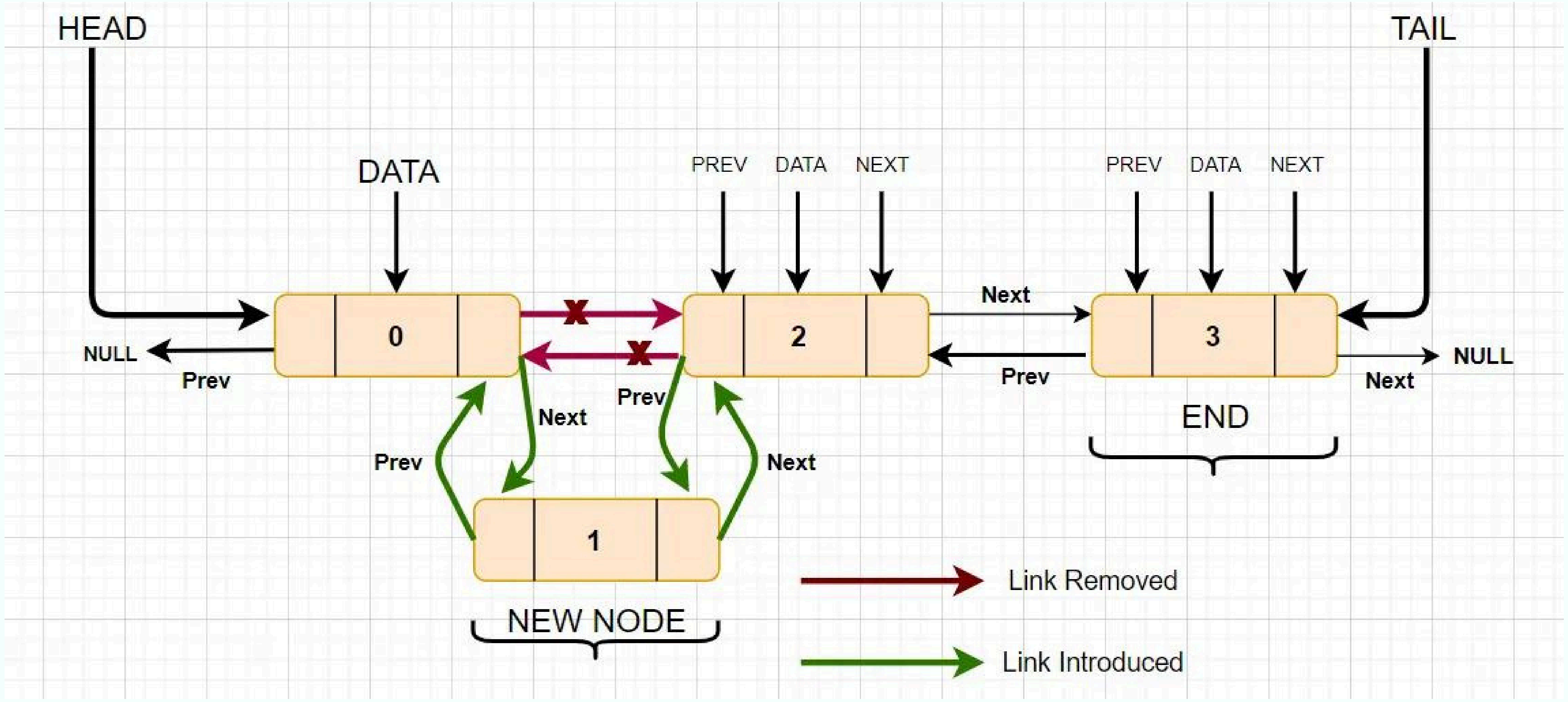
Operations on Linked list

Delete Node (given position)



Operations on Linked list

Insert node after



Question and Answer

Thank You