

Wednesday, 9
March 16

Crux

Lecture -13

Data Structures -1

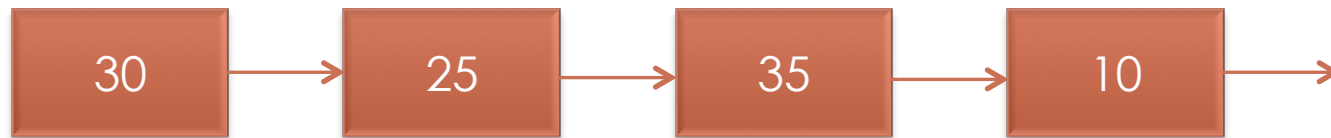
Linked Lists

Nidhi Agarwal



What are Data Structures?

What are Linked Lists?



Lets define our own Linked List

```
public class Node<T> {  
    T data;  
    Node<T> next;  
}
```

Head and Tail nodes

Basic operations over Linked List

- Taking Linked List as input from user
- Accessing next element
- Looping over Linked List
- Inserting into Linked List
- Deleting from Linked List

Lets do some problems

- Taking Linked List as input from user
- Print a Linked List

Your turn

- Print ith element of Linked List

Insertion at ith Position

Your turn

- Find length of Linked List
 - Iteratively
 - Recursively
- Delete the element at ith Position
- Implement Insertion/Deletion using Recursion

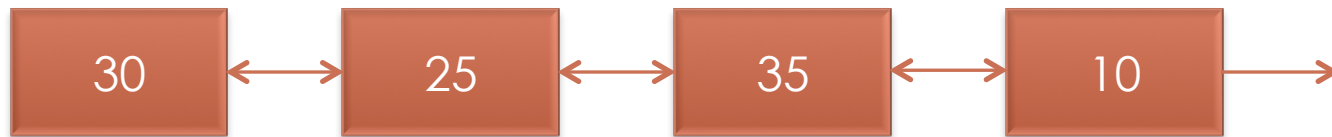
Benefits of Arrays over Linked List

- Random access to elements
- Fast iteration through the elements
- Very compact way to store data

Benefits of Linked List over Array

- Constant time insertion and deletion of elements
- Don't need to know the number of elements
- Insert elements in the middle of the list

Doubly Linked Lists



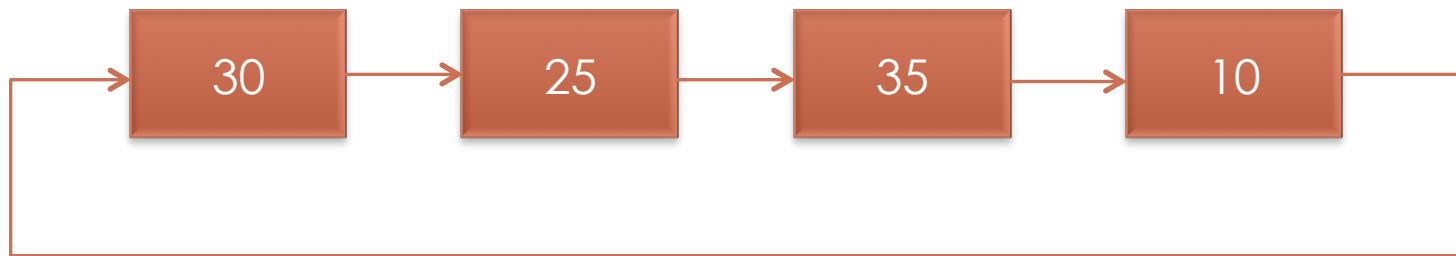
Implementation?

```
public class Node<T> {  
    T data;  
    Node<T> next;  
    Node<T> prev;  
}
```

Doubly LL vs Singly LL

- Faster to go back in the linked list
- Uses more memory

Circular Linked Lists



Lets try some problems

- Find an element iteratively and recursively
- Find mid point of a linked list
- Implement Bubble Sort

Your Turn

- Find 5th element from end without calculating length of Linked List
- Given two sorted linked lists merge them into a sorted linked list
- Implement merge sort
- Reverse a Linked List



Thank You !! 😊

Nidhi Agarwal

nidhi@codingblocks.com