

DSA through C++

Doubly linked list

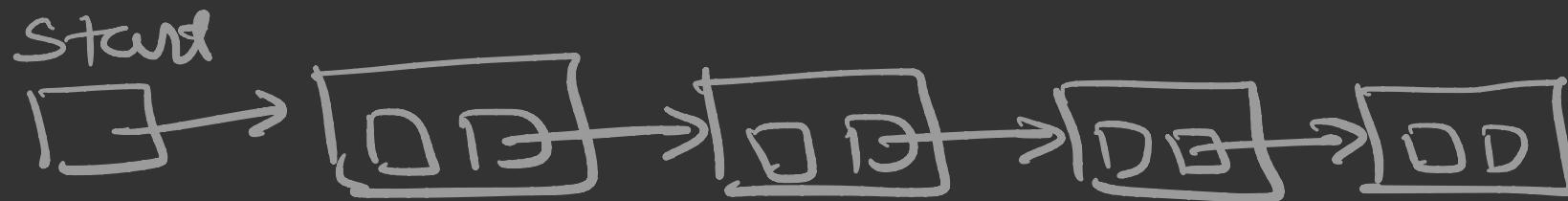


Saurabh Shukla (MySirG)

Agenda

- ① Shortcomings of singly linked list
- ② Doubly linked list
- ③ node
- ④ insertion
- ⑤ deletion

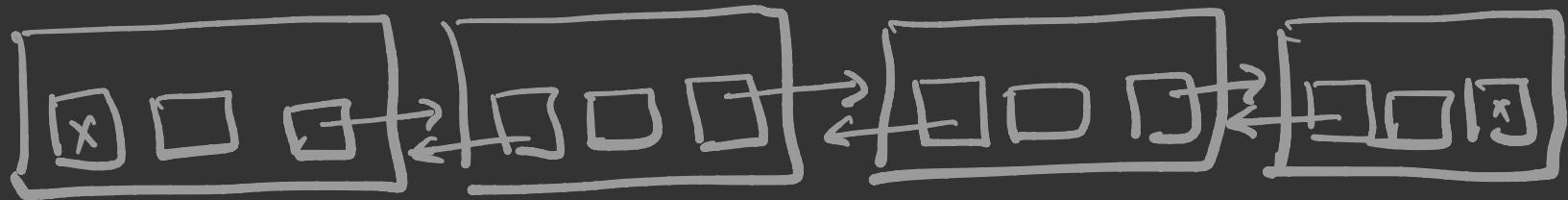
Shortcomings of Singly linked list



In SLL, you can move only in the forward direction

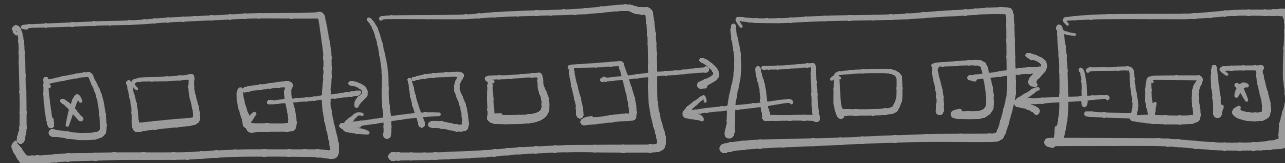
Doubly linked list

Start



node

Start



struct node
{

 node * prev;

 int item;

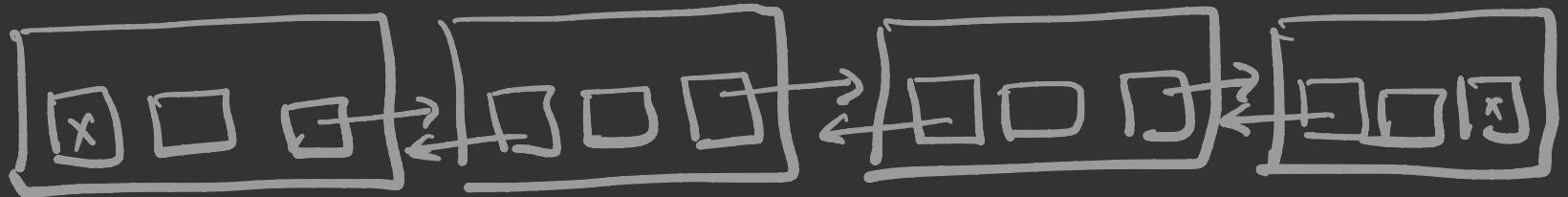
 node * next;

};



Insertion

Start

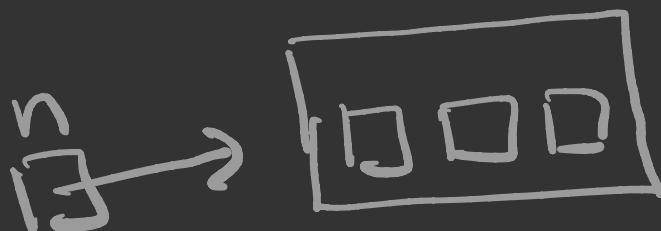
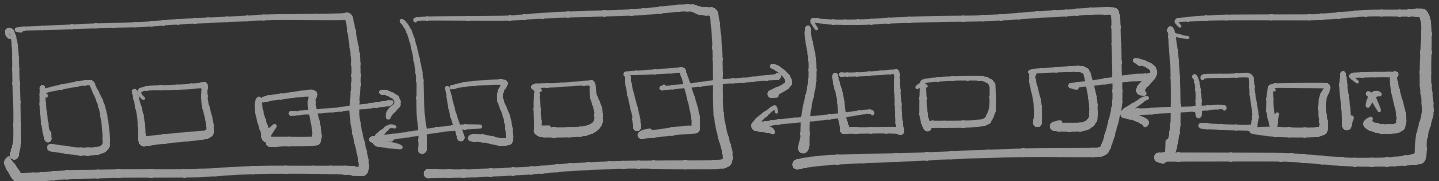


Insertion

- ① At First
- ② At Last
- ③ After a node

Insert as a first node

Start

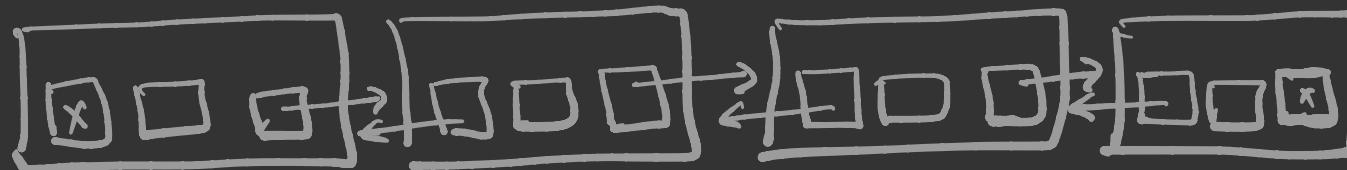


```
node *n = new node;  
n->item = data;  
n->prev = NULL;  
n->next = Start;  
start->prev = n;
```

```
start = n;
```

Insert at last

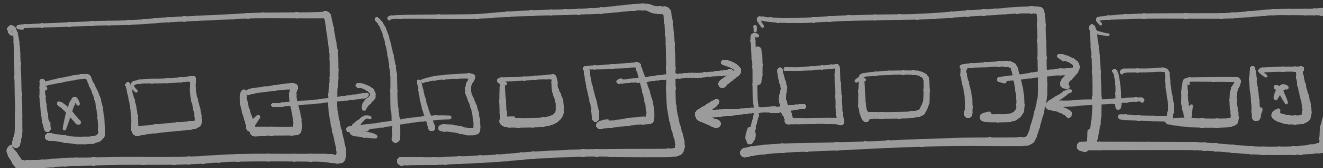
Start



```
node *n = new node;  
n->item = data;  
n->next = NULL;  
t = Start;  
while (t->next != NULL)  
    t = t->next;  
n->prev = t;  
t->next = n;
```

Insert After a node

Start



```
node *n = new node;  
n->item = data;  
n->prev = t;  
n->next = t->next;  
t->next->prev = n;  
t->next = n;
```

Deletion

- ① Delete first node
- ② Delete Last Node
- ③ Delete specific node