

DSA through C++

Circular doubly linked list

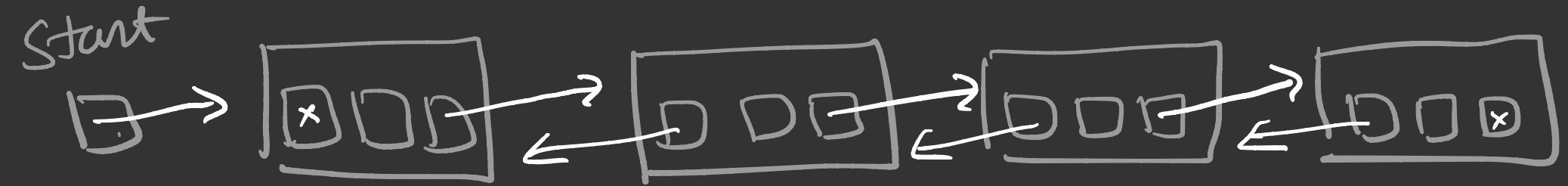


Saurabh Shukla (MySirG)

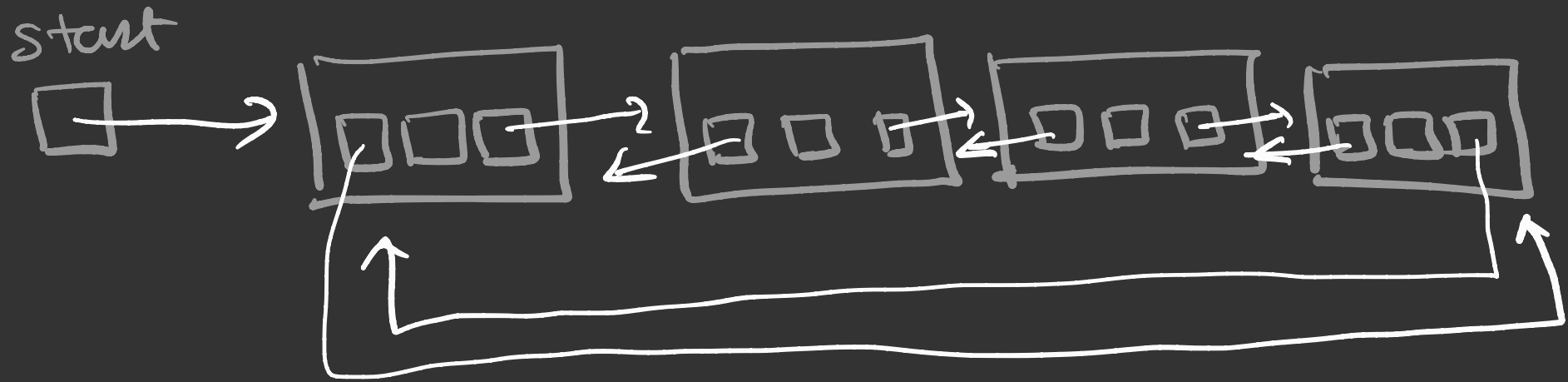
Agenda

- ① Not utilizing next and prev pointers in DLL
- ② Circular doubly linked list
- ③ node
- ④ insertion and deletion

Not utilizing next and prev pointers in DLL



Circular Doubly Linked List



node

```
struct node
```

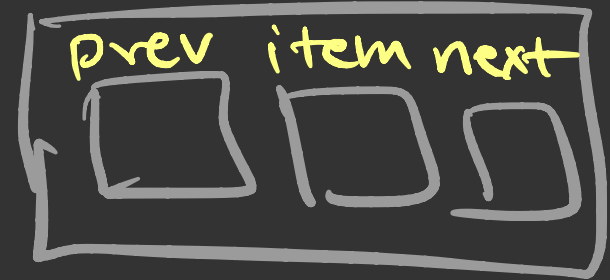
```
{
```

```
    node *prev;
```

```
    int item;
```

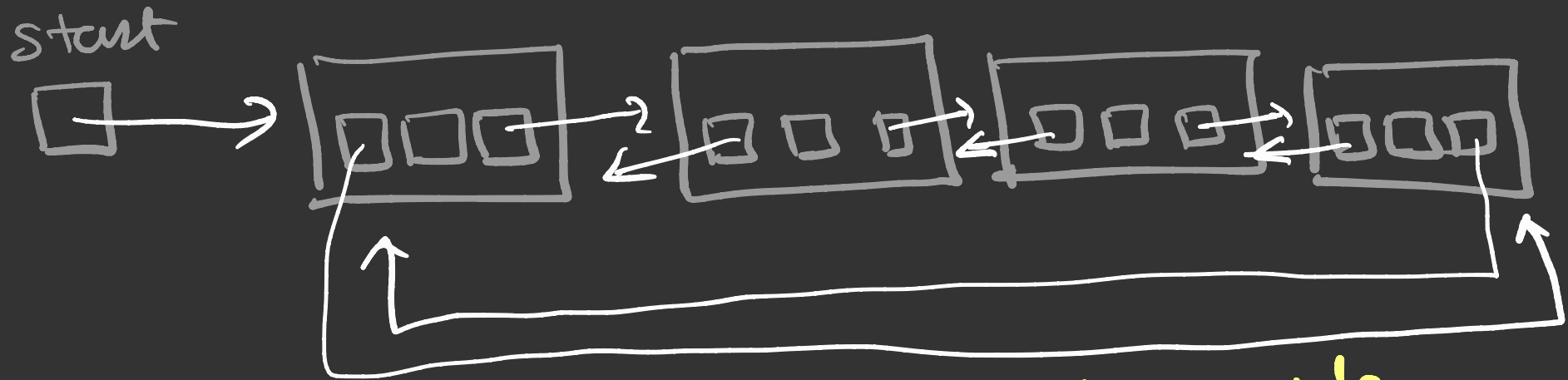
```
    node *next;
```

```
}
```



Insertion in CDLL

node *n = new node;
n->item = data;



First node

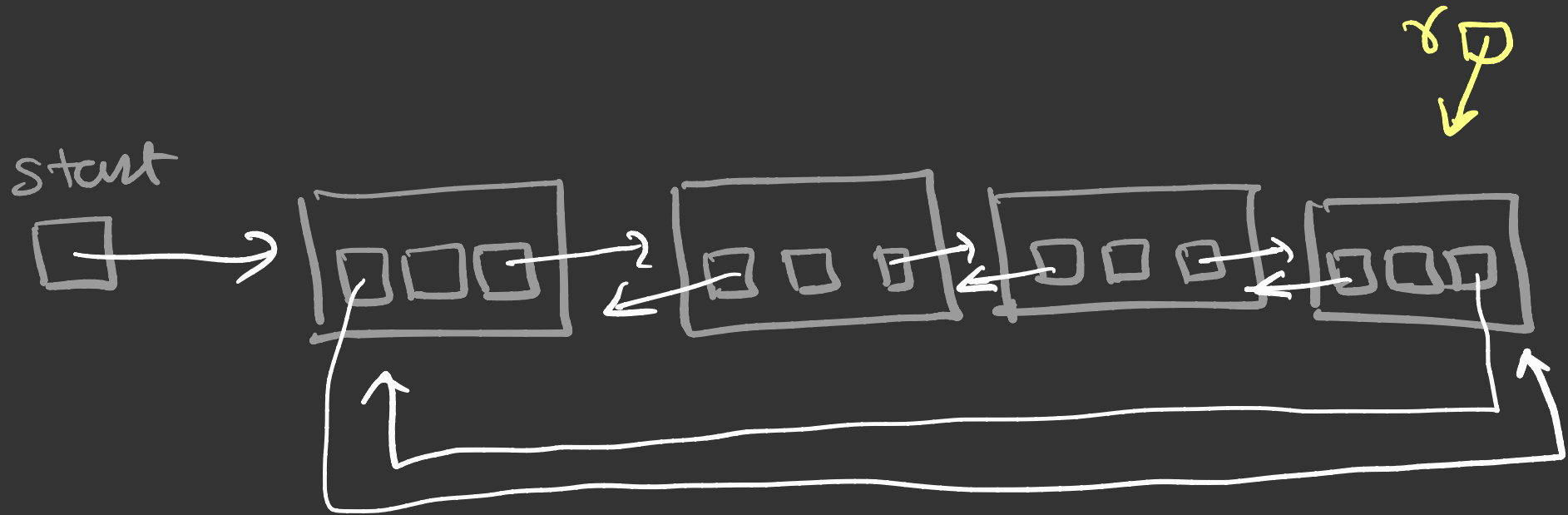
$n \rightarrow \text{prev} = \text{start} \rightarrow \text{prev};$
 $n \rightarrow \text{next} = \text{start};$
 $n \rightarrow \text{next} \rightarrow \text{prev} = n;$
 $n \rightarrow \text{prev} \rightarrow \text{next} = n;$
 $\text{start} = n;$



last node

$n \rightarrow \text{prev} = \text{start} \rightarrow \text{prev};$
 $n \rightarrow \text{next} = \text{start};$
 $n \rightarrow \text{prev} \rightarrow \text{next} = n;$
 $n \rightarrow \text{next} \rightarrow \text{prev} = n;$

Deletion in CDLL



```
x = start;  
start → prev → next = x → next;  
x → next → prev = x → prev;  
start = x → next;  
delete x;
```

```
x → prev → next = start;  
start → prev = x → prev;  
delete x;
```