→ TA office hours (POC: Dhruv)

→ [9:30 - 11 pm
micro]

→ 20% assignments
+
Mock Interview (AI)
(50%, 100%, 0%,)

- → Hints / Sol approx. / Verdict
- → Companion TA
- → Peers
- → TAs

Bit Manipulation (Practice)

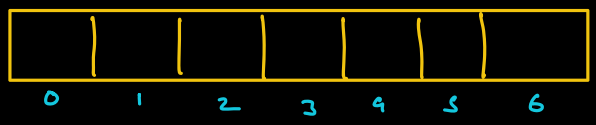
→ Results
- → Caught Cheat → (F)
- → Re-exam (1st & 2nd Mock)

  ↳ for students who have been flagged by the system

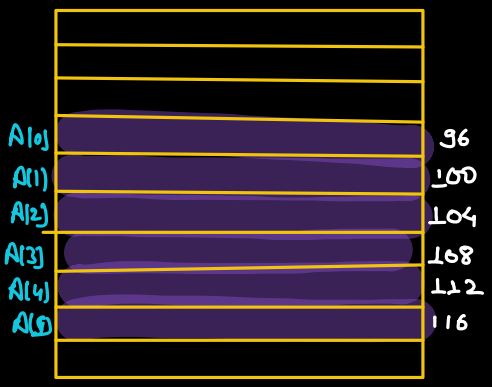
→ Quizzes
- ? = 80% → 100%
- ? = 50% → 50%

Arrays (Array list / vector)

int:



O(1) → Random Access



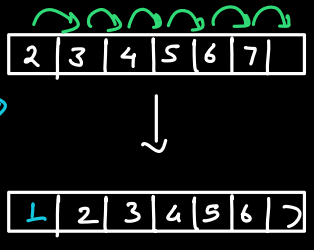
A[i] → O(1)

A[i] = x; O(1)

Add of A[0] + i x Size-of-data type

int[] B = new int[4];

Insert



At start

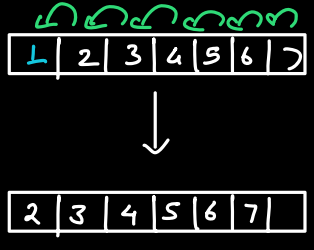
O(N)

↓

O(1) → LL

K^th Pos O(N)

Delete



O(N)

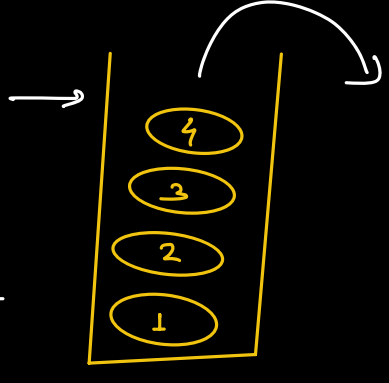
↓

O(1) → LL

O(N)

LIFO

FIFO (Queue)



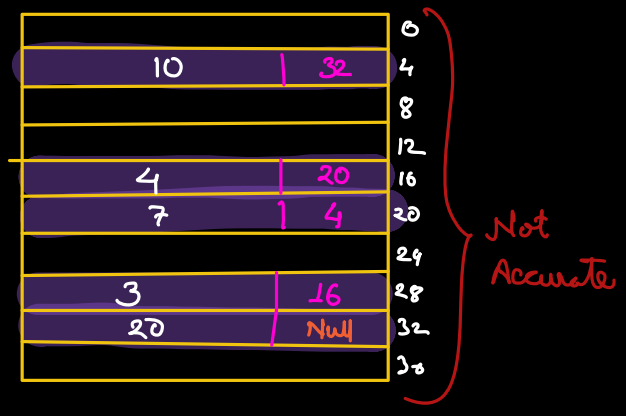
Inverted Index

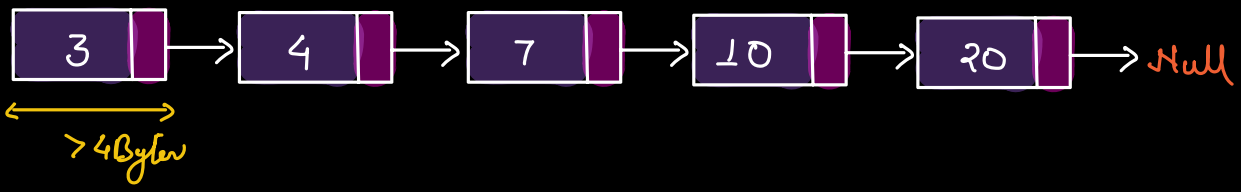
Linked list

3 , 4 , 7 , 10, 20

→ Why ?

→ ~~Right to left ?~~ ×



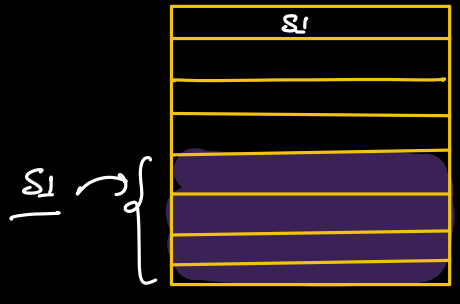



```
class Node {
    int data;
    Node next;
    public Node (int val){
        this.data = val;
        this.next = null;
    }
}
```

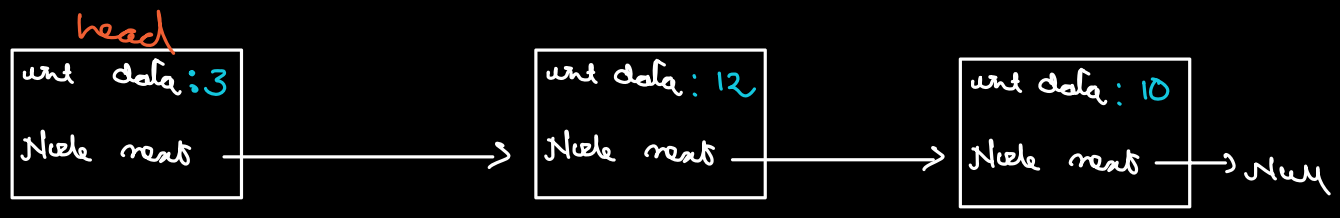
Reference

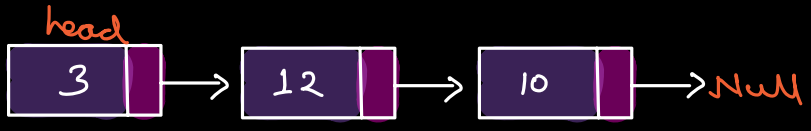
```
Student S1;
S1 = new Student("_", "_");
```

1115




```
class User {
    User mentor;
    User TA;
    User instructe;
}
User u1 = new Use();
```






```
Node head = new Node(3);
head.next = new Node(12);
head.next.next = new Node(10);
```

```
void fn(int[] A){
    ---
}
```

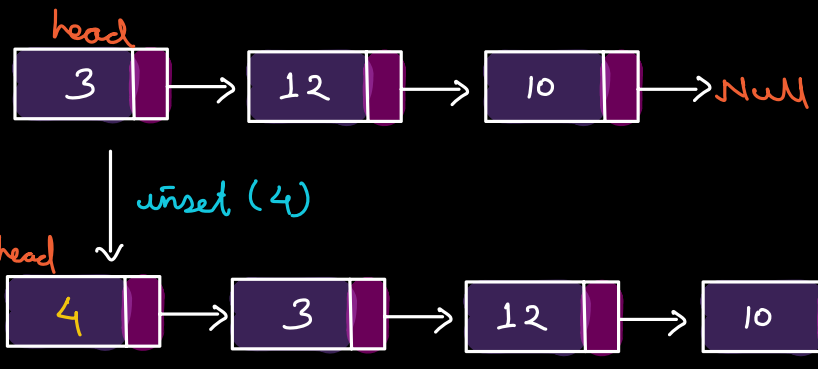
```
main(){
    int A[] = {3, 4, 5}
    fn(A);
}
```

Ref to A[0]

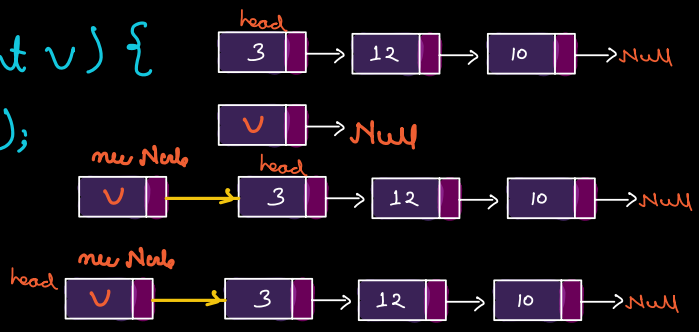
```
void fn(Node head){
    ---
}
```

```
main(){
    Node head = new Node(3);
    head.next = new Node(12);
    head.next.next = new Node(10);
    fn(head);
}
```

Insert at Start

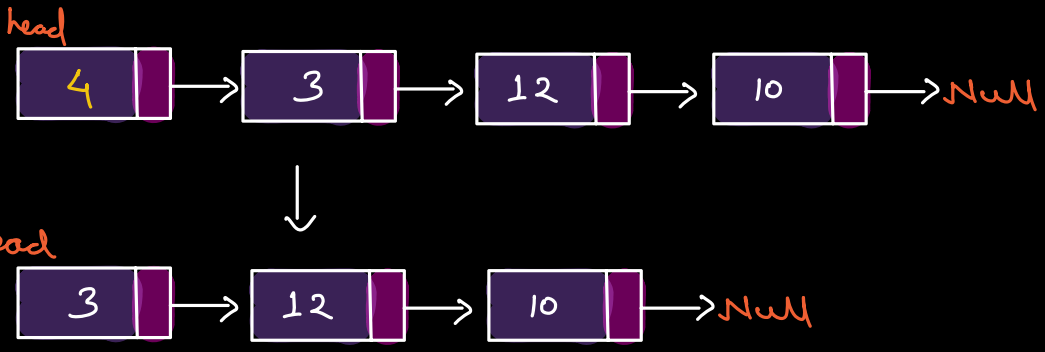



```
Node insertAtStart (Node head, int v) {
    Node newNode = new Node (v);
    newNode.next = head;
    head = newNode;
    ret head;
}
```

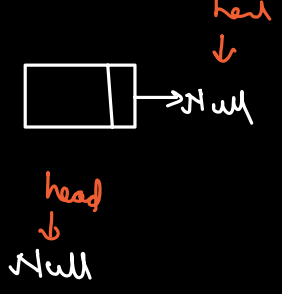


TC: O(1)

Delete from the Start




```
Node deleteFromStart (Node head){
    if(head == null) ret null;
    head = head.next;
    return head;
}
```



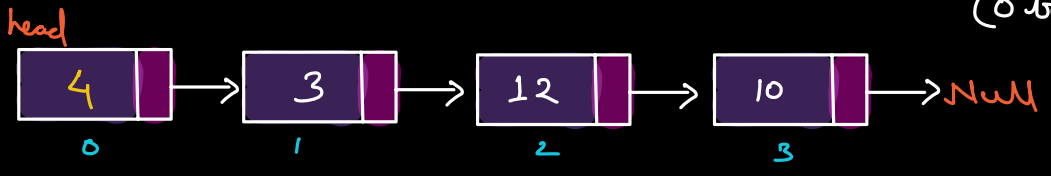
NullPointerException
- null.data
- null.next

TC: O(1)

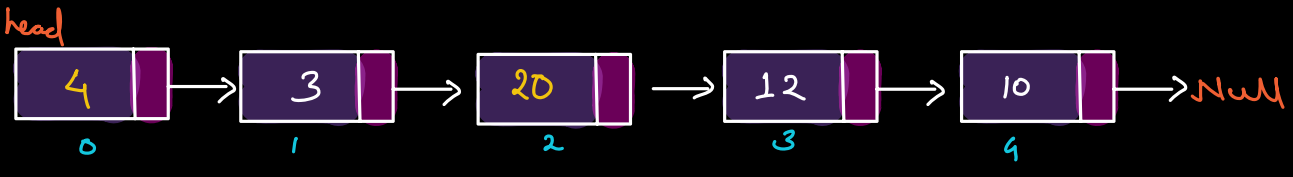
Cache

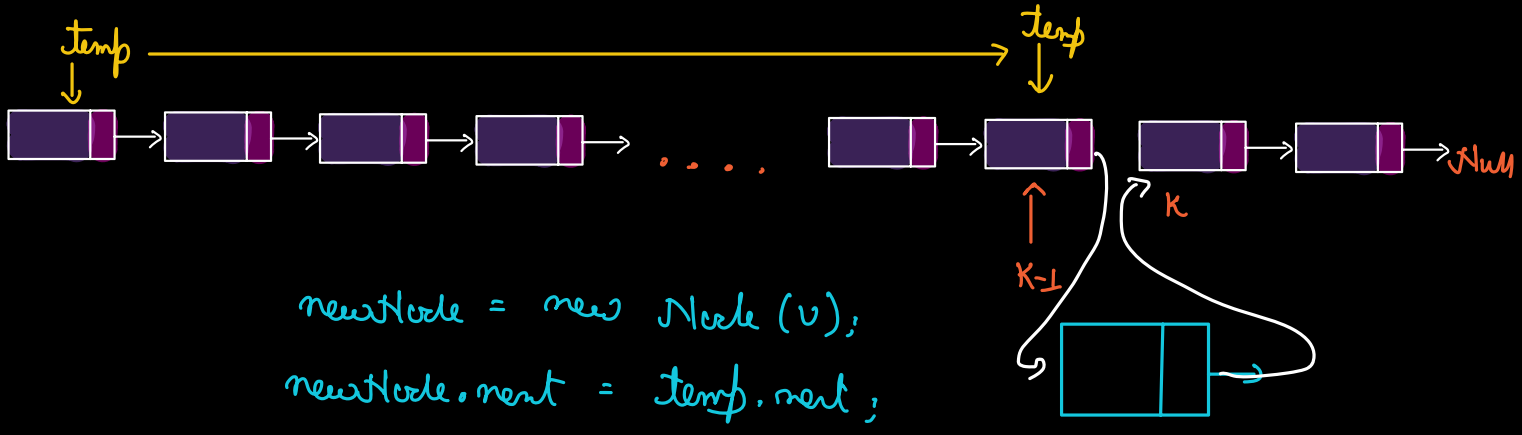
LRU

Q Given a LL. Insert an element at K^{th} position. (0 based)



K = 2
Val: 20






```
newNode = new Node(v);
newNode.next = temp.next;
temp.next = newNode;
```

Q Length of LL

```
int a = 5;
a++;
```

```
int length(Node head){
    Node temp = head;
    int len = 0;
    while(temp != Null){
        len++;
        temp = temp.next;
    }
    return len;
}
```

T-II

$\tau = 70\%$