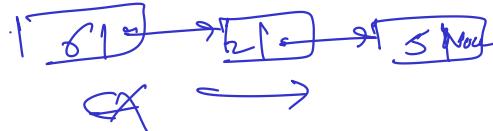


Doubly linked list



Class node

```

class node {
public:
    int data;
    node *next;
    node *prev;
    node (int val)
}
  
```

list ✓

→ Array based

→ fast.

→ size (fixed)

→ linked list

→ slow

→ size (flexible)

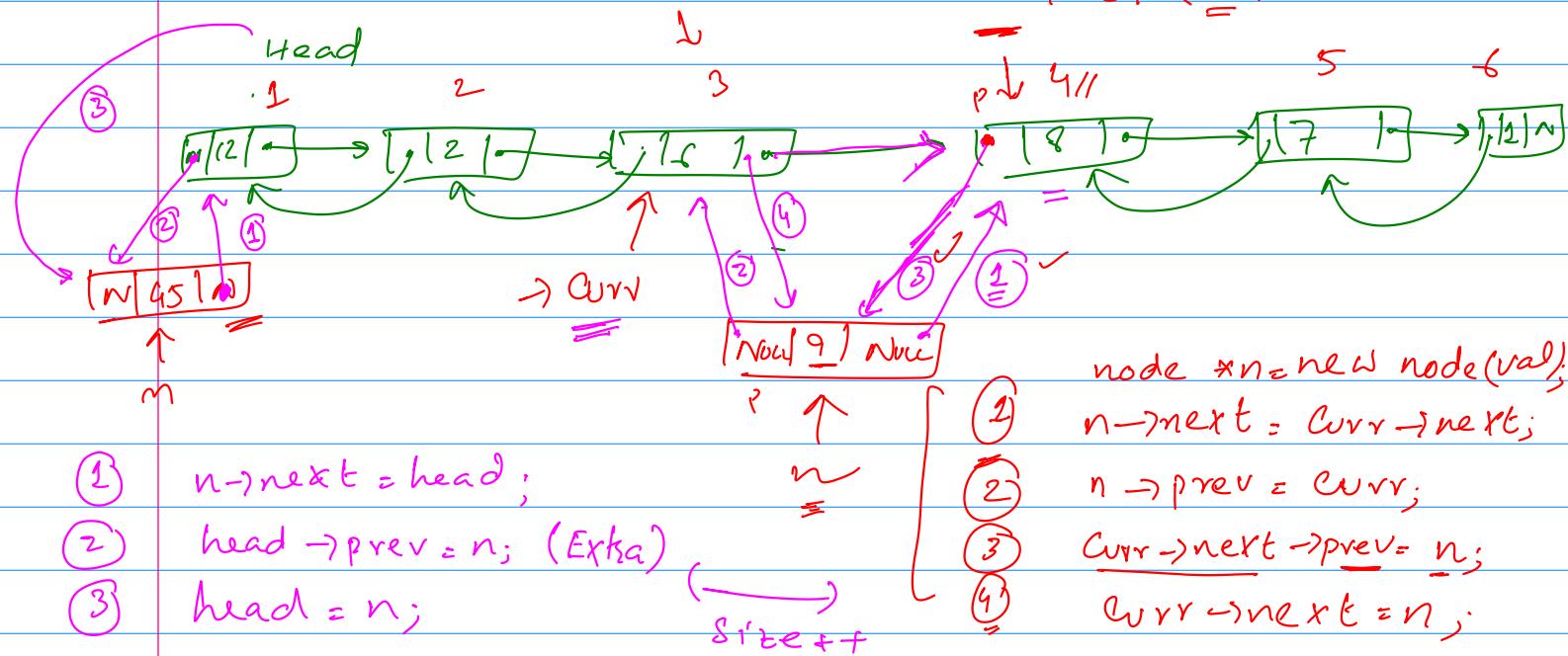
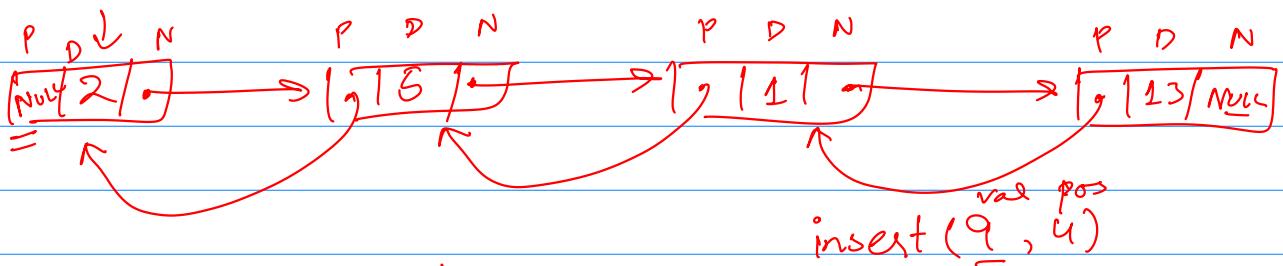
data = val;

next = NULL;

prev = NULL; ✓

}; head ✓

size = 4.



Josephus problem

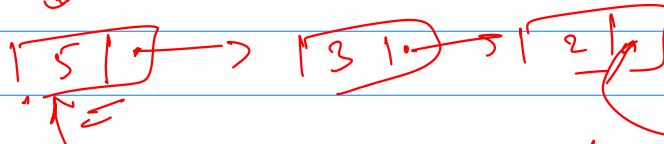
circular

10 nodes

100

$$= \times$$

1



$n \rightarrow \text{next} = \text{curr} \rightarrow \text{next}$

2
1
3
4

longer

1

int

joshlepos (int m > int n) {

leader

$$m = 3_{11}$$

no. of nodes

delete

return *head;

Circularly
Connect

Doubly

① ~~head~~

② $n \rightarrow \text{next} = \text{head};$

③ $\text{head} \rightarrow \text{prev} = n;$

④ $\text{head} \rightarrow \text{prev} = \underline{n};$

⑤ $\underline{\text{head}} = n;$

curr

node * curr = head -> prev;