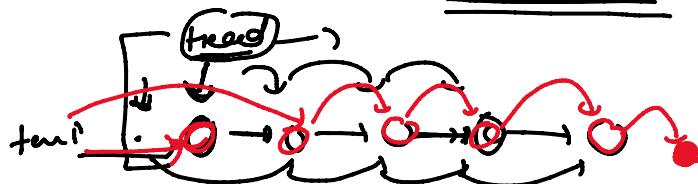
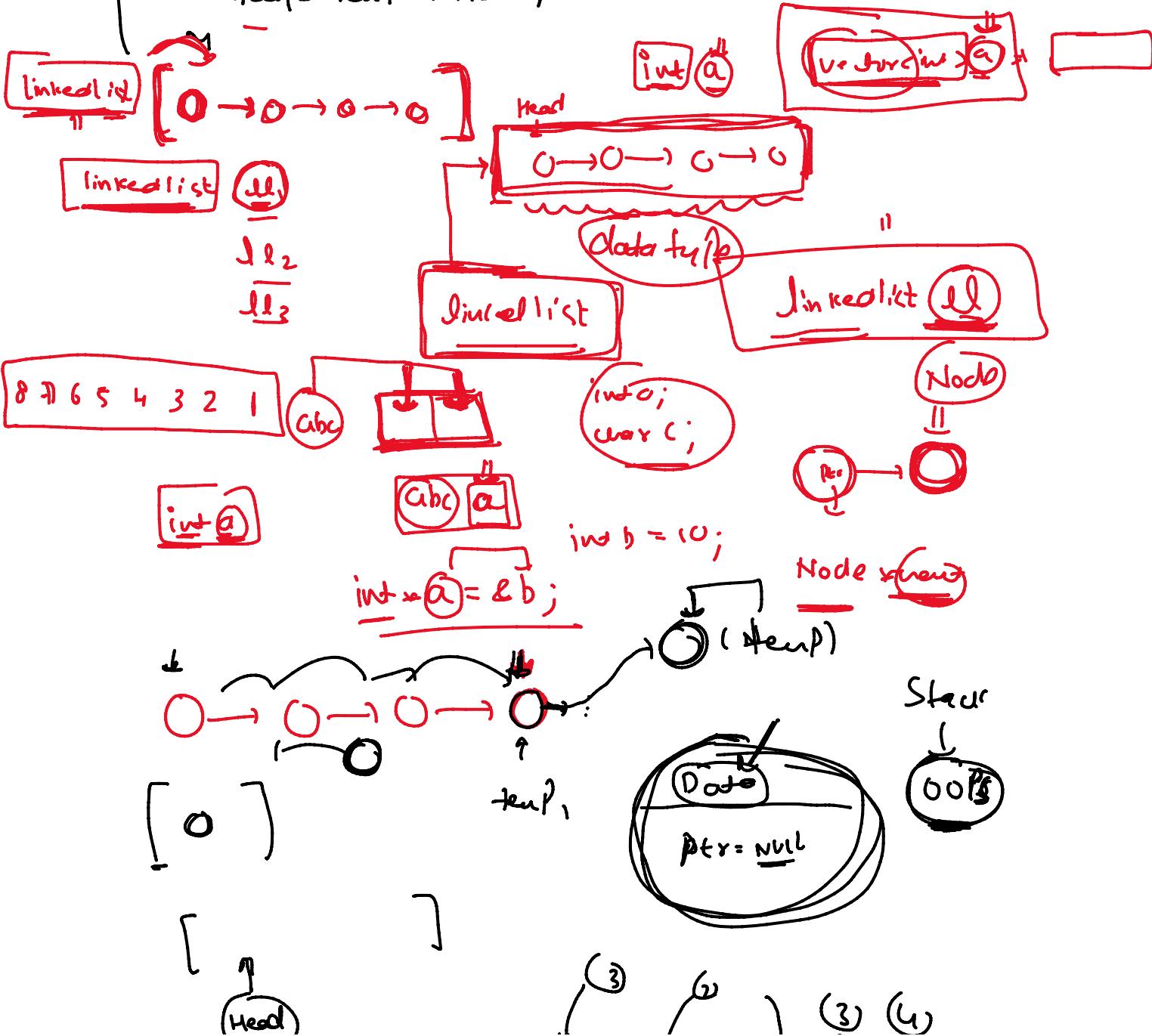
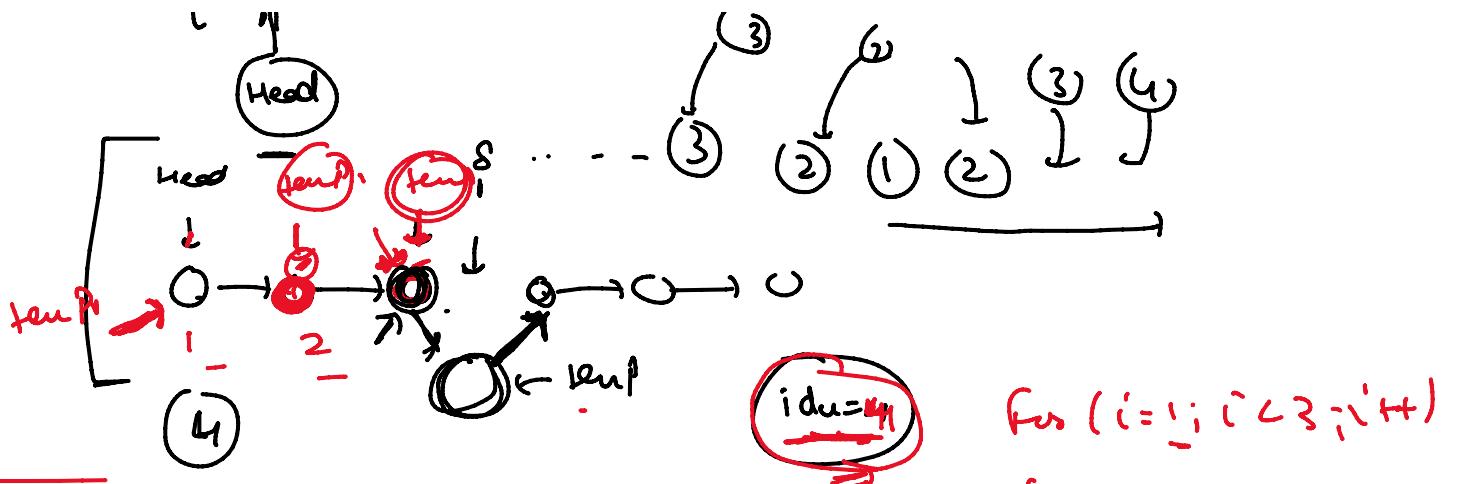


CLASS-50

$$\underline{\text{Node} * \text{temp} = \text{head};}$$

```
while ( temp != NULL )
    & ( cout << temp->data << " " );
    temp = temp -> next;
```





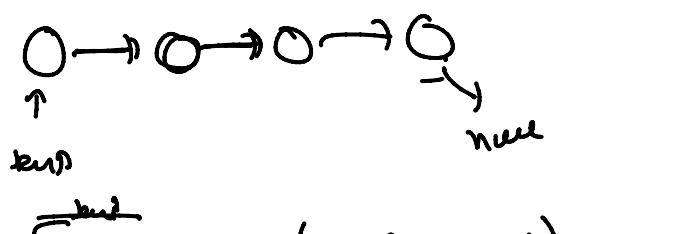
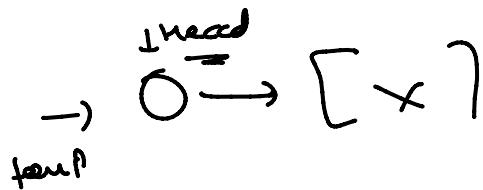
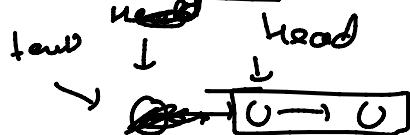
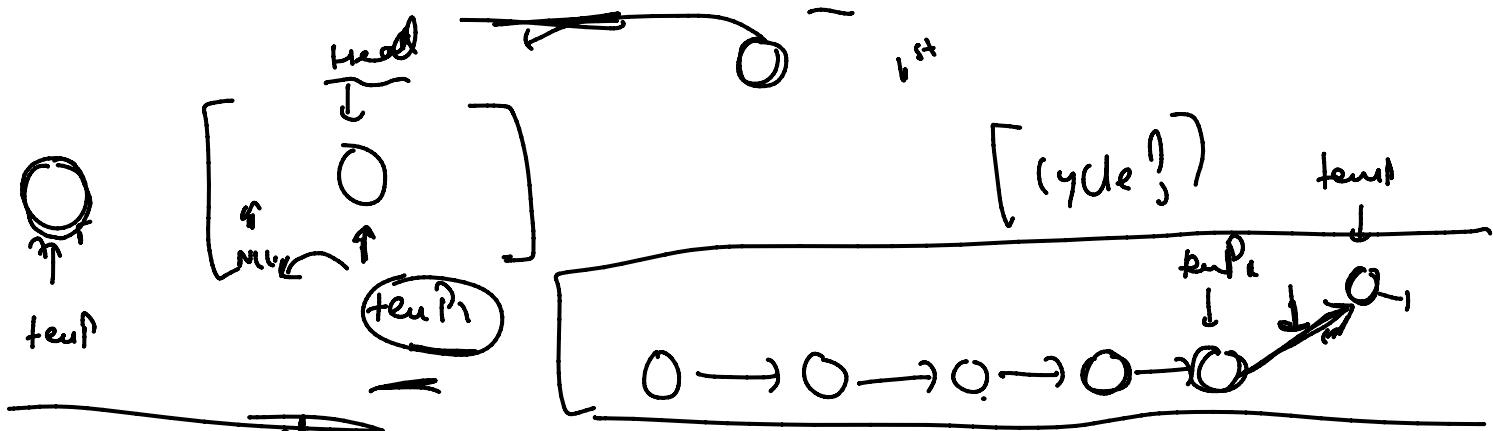
For ($i = 1$; $i < 3$; $i++$)

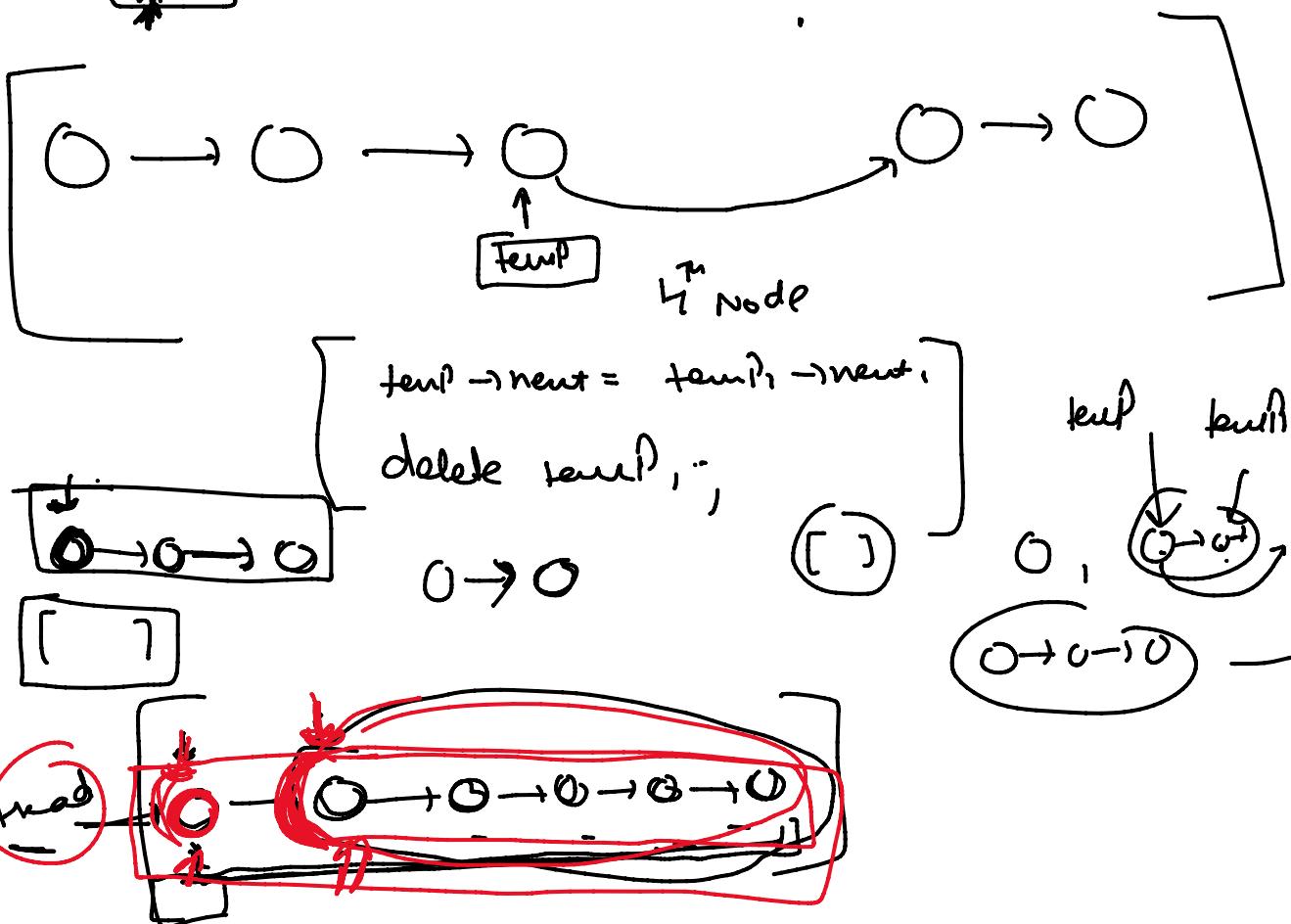
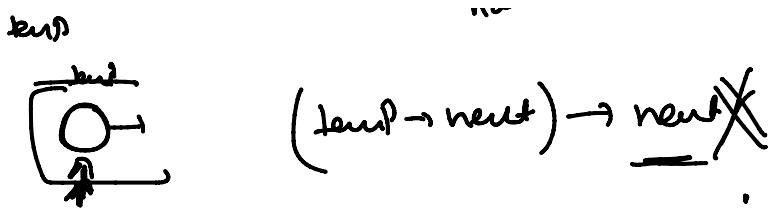
{
 $tempP_i = temp_i \rightarrow next_i$

$temp_i \rightarrow next = temp_i$
 $temp_i \rightarrow next = temp$

$i = 4$

$tempP_i = temp_i \rightarrow next$





Problem :- Find length of linked list starting from head

Subproblem (Head → next);

Subproblem Manually

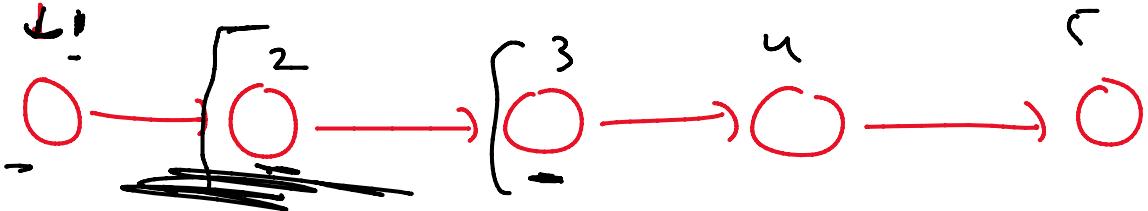
1 + subproblem (Head → next)

```
int findlength ( Node* head )
{
    if ( Head == NULL )
        return 0; // Base case
    else
        return 1 + findlength ( Head → next );
}
```

// Base case?
[]

return l + findlength (head → next);

head



findlength (Head)

findlength (1) → $l+1 = 1$

findlength (2) → $2+1$

findlength (3) = $1 + \underline{\text{find}(4)}$

findlength (4) = $1 + \underline{\text{find}(5)}$

findlength (5) = $1 + \underline{\text{find}(6)}$

findlength (6)