

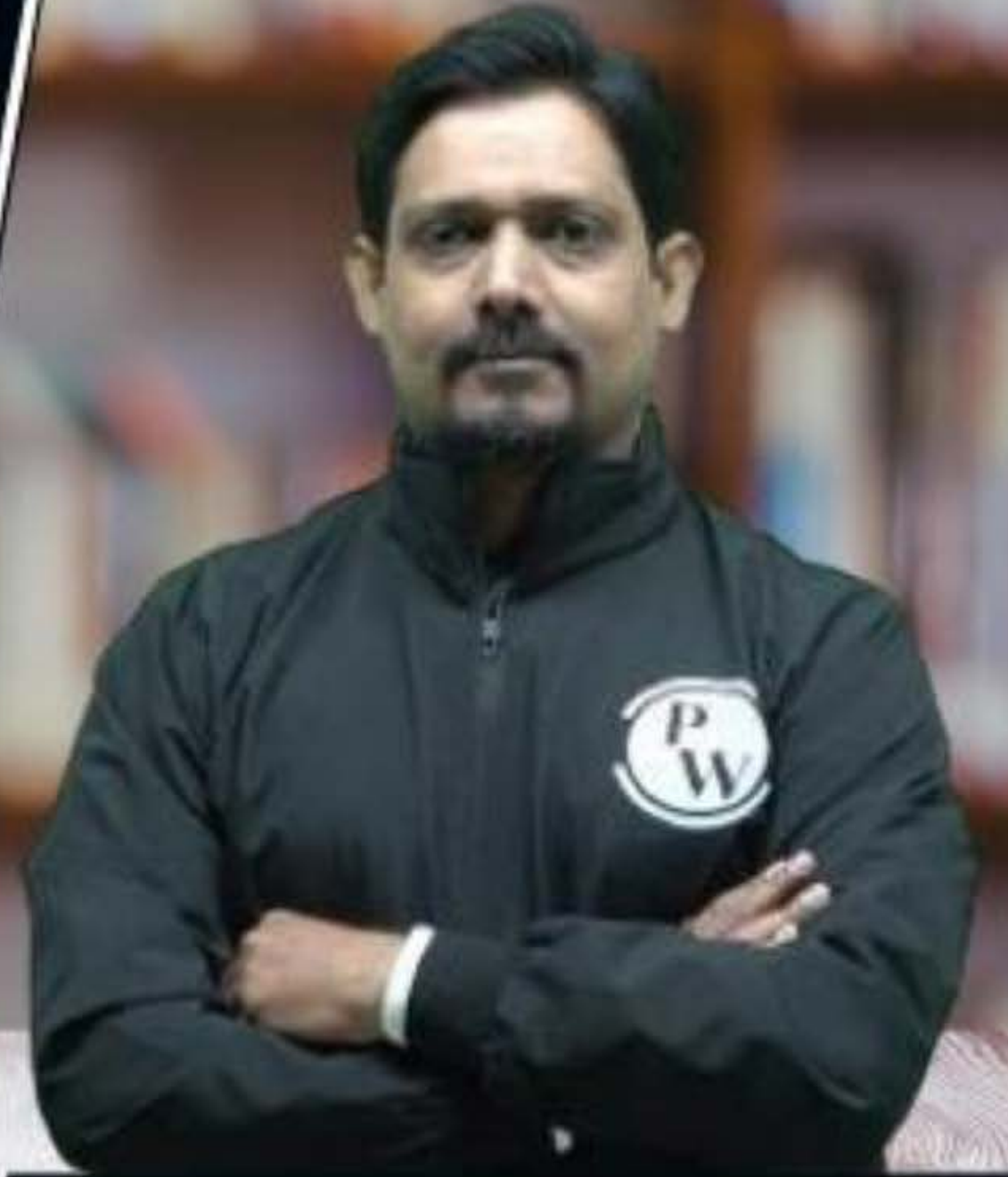
Computer Science & IT

Data Structure & Programming



Linked List

Lecture No. 02



By- Abhishek Sir

Recap of Previous Lecture



Topic

Linked List

Topic

Node, getnode, build123

Topic

Count & display

Topic

Topic

Topics to be Covered



Topic

Insertion, begin, end

Topic

Delete end

Topic

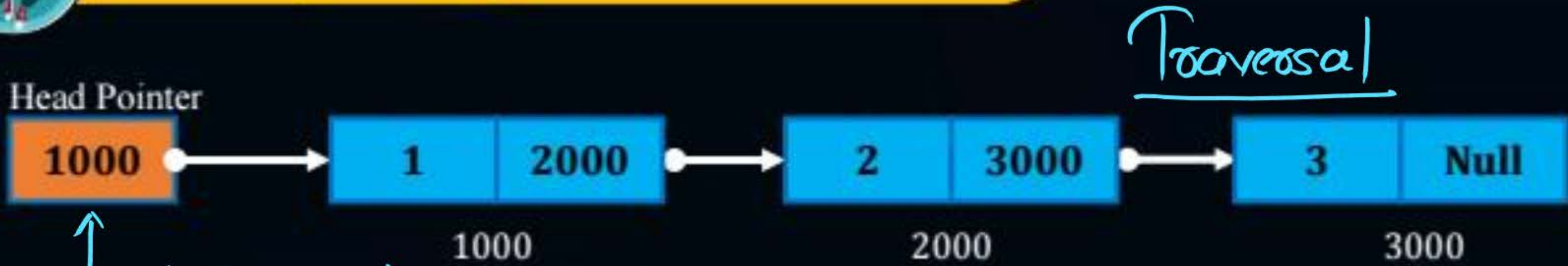
Recursive function

Topic

Topic



Topic : Single Linked List



Traversal

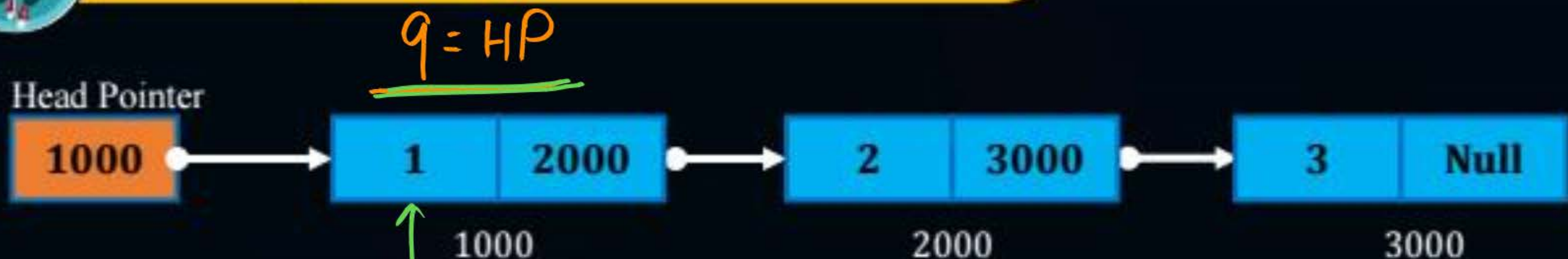
↑ HP - global variable

Add a Node at beginning of list

1. traversal Not required
2. Head pointer will modify??

yes

Topic : Single Linked List



$p \rightarrow \text{next} = q;$

1. Create a Node with x
2. Check empty
3. Modify Link of New Node



Topic : Single Linked List



Is this function depends
upon size of linked List

Independent of size

Constant time $O(1)$

Constant

```
void Addbegin(int x){
```

```
Node * p = getnode(x);
```

```
if (HP == NULL){
```

```
    HP = p;
```

```
    return;
```

```
}
```

```
p → next = HP;
```

```
HP = p;
```

```
return;
```

```
}
```




Topic : Single Linked List



Add a Node at the end of list.

* Last Node next part is NULL

Logic -

- * Traversal Needed $q \rightarrow next = p$
- * Head pointer will modify or Not



Topic : Single Linked List



```
while (q → next != NULL)
```

```
    q = q → next;
```

```
    q → next = p;
```

```
    return;
```

```
}
```

```
void addend (int x) {
```

```
    Node *q = HP;
```

```
    Node *p = getnode(x);
```

```
    if (HP == NULL) {
```

```
        HP = p;
```

```
        return;
```

```
    }
```

Is this depends

upon size of list ??

yes

$O(n)$



Topic : Single Linked List



delete Last Node

* traversal Needed or Not

* Second Last Node address Needed

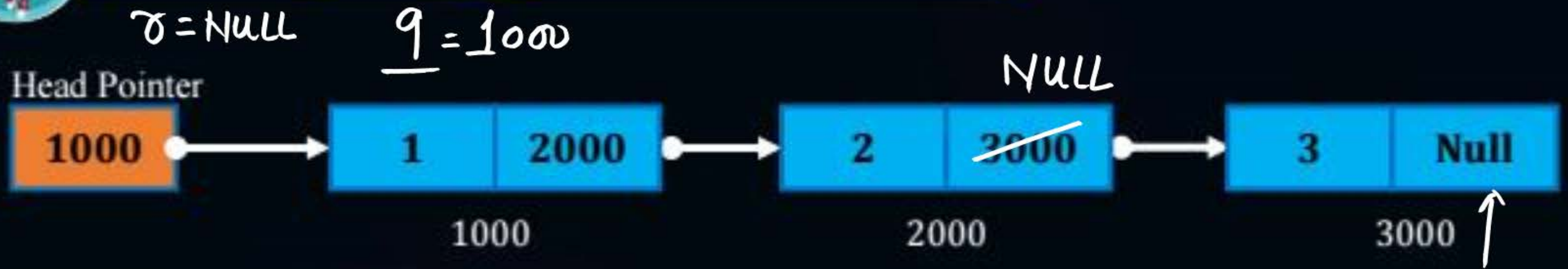
* Two pointer Iteration

NULL
pointer
close fence

9 → next → next



Topic : Single Linked List



$\delta = \underline{1000}$

$q = 2000$
 $\delta = 2000$

$q = 3000$





Topic : Single Linked List



```
void Deleteend ( ) {  
    Node *q = HP, *r = NULL;  
    if (HP == NULL)  
        return;  
    while (q->next != NULL) {  
        r = q;  
        q = q->next;  
    }
```

```
    if (r == NULL) {  
        HP = NULL;  
    }  
    else  
        r->next = NULL;  
    free(q);  
}
```

free takes
address of
a Node
to be
deallocated
from Heap



Topic : Single Linked List

if memory allocated & Not deallocated

then it results in Memory Leak.

↑
Not an Error



Topic : Single Linked List

Single Node Last deletion

1 2000 2, 3000 3 NULL
1000 2000 3000

$\delta = \text{NULL}$



while($q \rightarrow \text{next} \neq \text{NULL}$)

$\delta \rightarrow \text{next} = \text{NULL}$

if δ is NULL

$\delta \rightarrow \text{next}; \text{NULL} \rightarrow \text{next}$

NULL pointed δ
dereferencing

error



Topic : Single Linked List

Suppose the linked list contains $1 \rightarrow 2 \rightarrow 3$
and following function is called with Address of first Node;
HP is global.



Topic : Single Linked List

1 → 2 → 3

```
void foo(Node *q) {
```

output of function.

```
    if(q) {
```

```
        foo(q → next),
```

```
        printf("%d", q → data);
```

```
    }
```

```
}
```



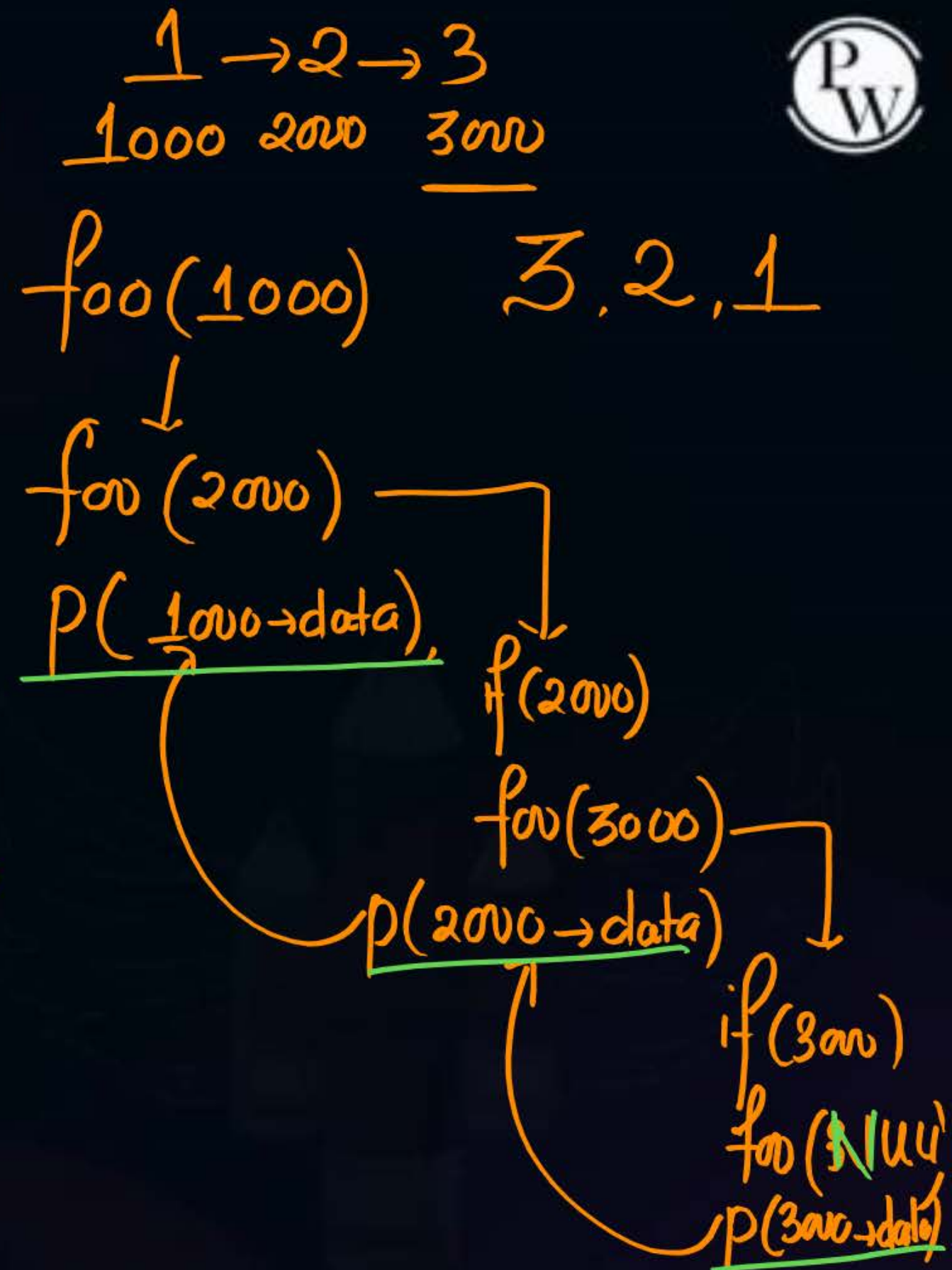
Topic : Single Linked List



```
void foo (Node *q) {
```

It
will print
Linked List
Reverse
order
}

```
    if (q) {  
        foo(q->next);  
        printf("%d", q->data);  
    }
```





Topic Question

If the list contain the value 1→2→3 then what is the Output following function.

```
void foo (Node *ptr){  
    if (ptr){  
        foo (ptr -> link);  
        printf ("%d", ptr ->data);  
        foo (ptr->link);  
        return;  
    }  
}
```

No. of values printed?

7

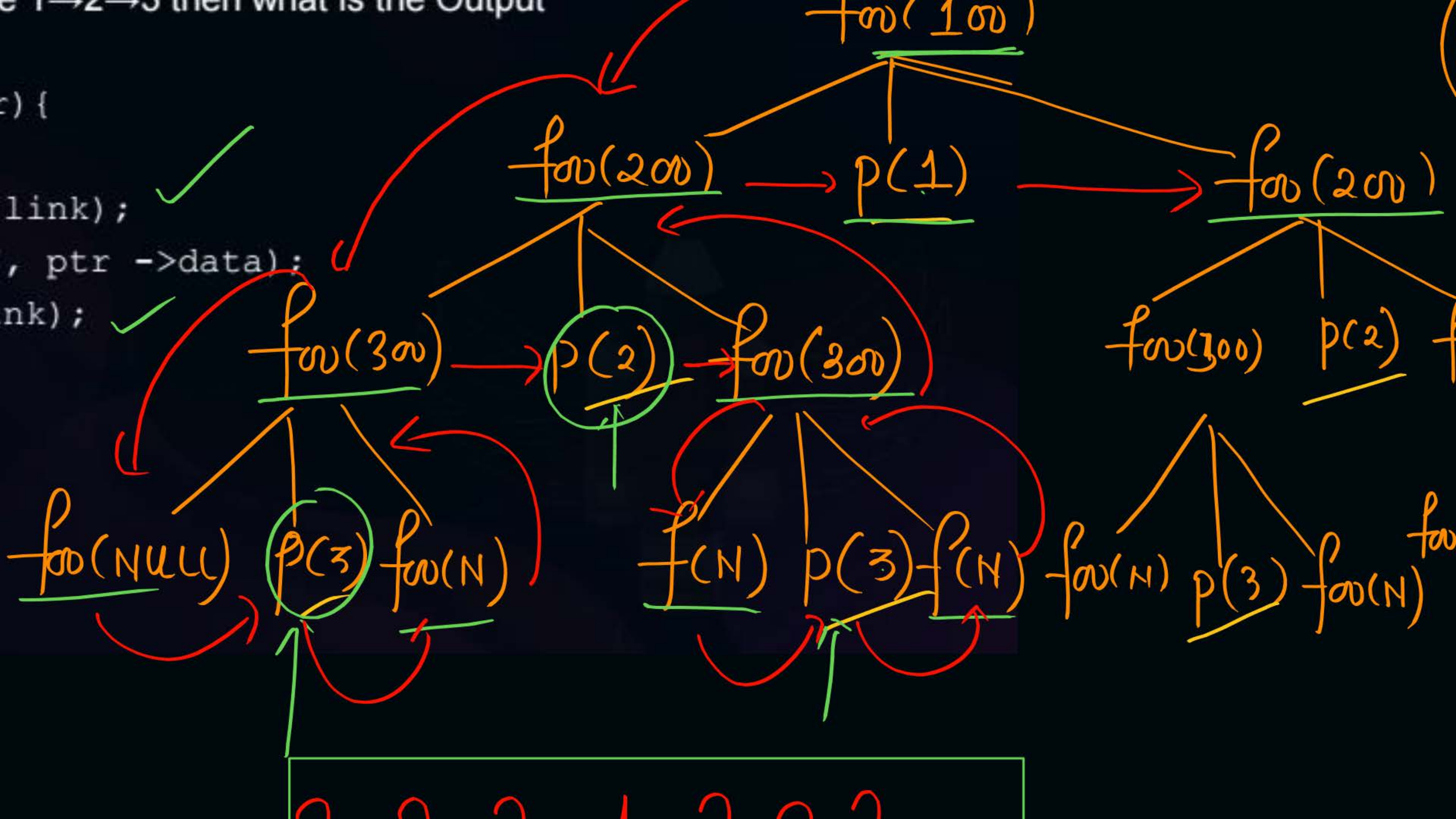
1 → 2 → 3 then what is the Output

c) {

link);

, ptr ->data);

nk);



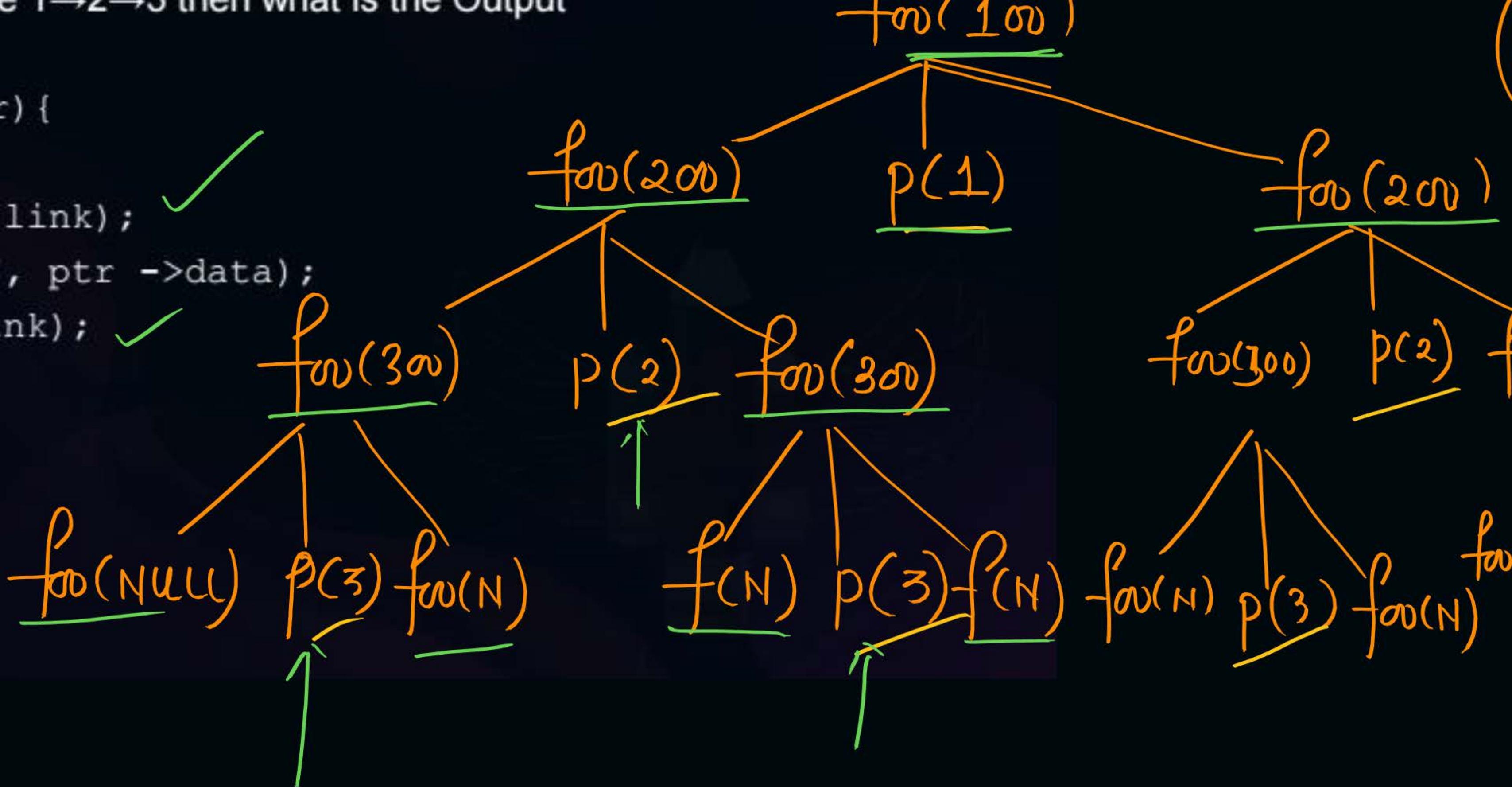
1 → 2 → 3 then what is the Output

c) {

link); ✓

, ptr ->data);

nk); ✓





2 mins Summary



Topic

Insert, begin, end

Topic

Delete end

Topic

NULL pointer dereferencing

Topic

Topic

THANK - YOU