CS & IT ENGINEERING

Data structure & Programming

Linked List

Lec- 01

By- Pankaj Sharma sir

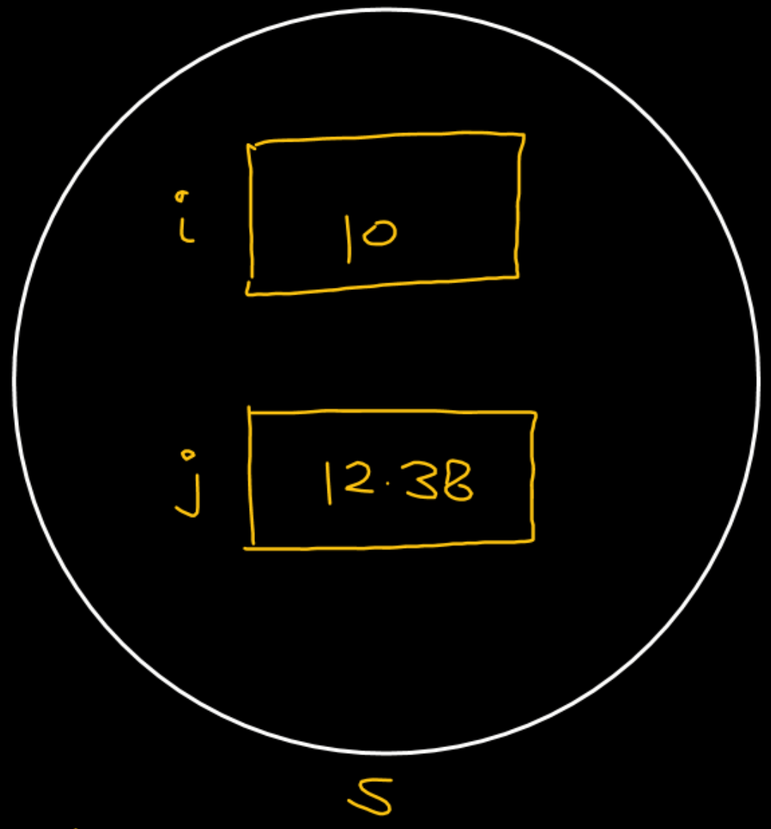
Linked List -1

```
struct my_struct {
    int i;
    float j;
};
```

diff types

```
void main(){
    struct my_struct s;
}
```

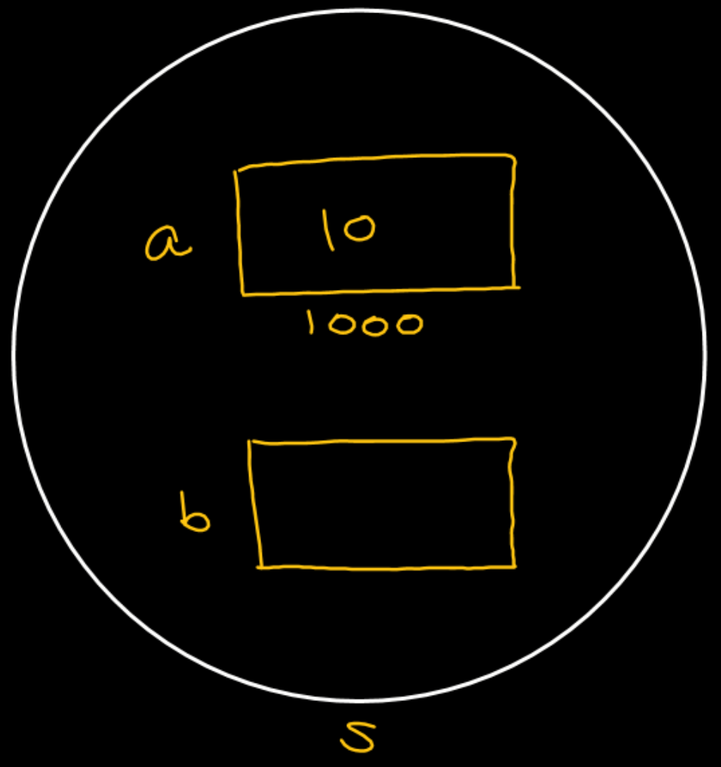
=




```
s.i = 10;
s.j = 12.38;
```

```
struct my_struct {
        int a;
        int *b;
        };
void main(){
        struct my_struct s;
        s.a = 10;
        s.b =
```

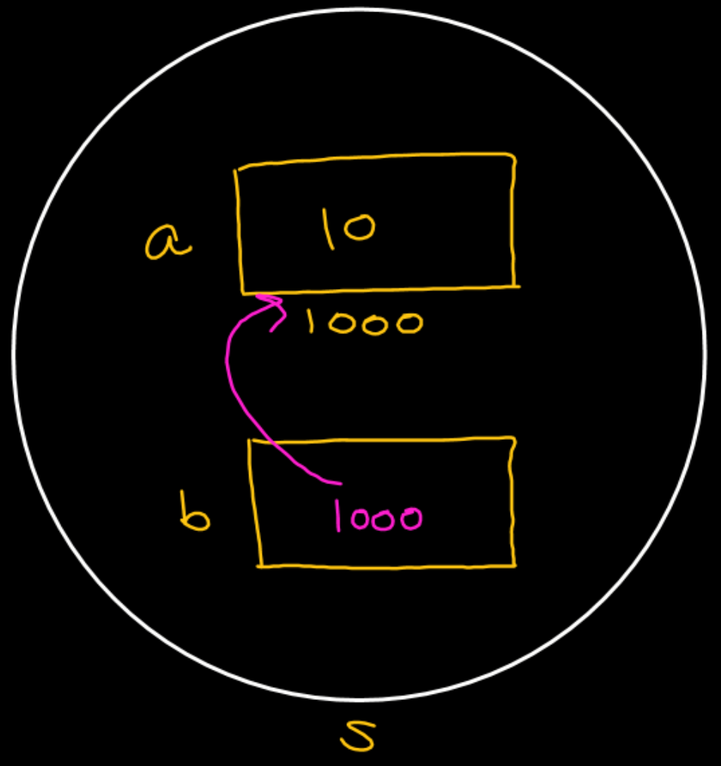
address of some integer var.



```
struct my_struct {
    int a;
    int *b;
};
```

→ member can be a pointer variable

```
void main(){
    struct my_struct s;
    s.a = 10;
    s.b = &s.a;
```



```
struct Pankaj{
    int a;
    int *ptr;
};
```

✓

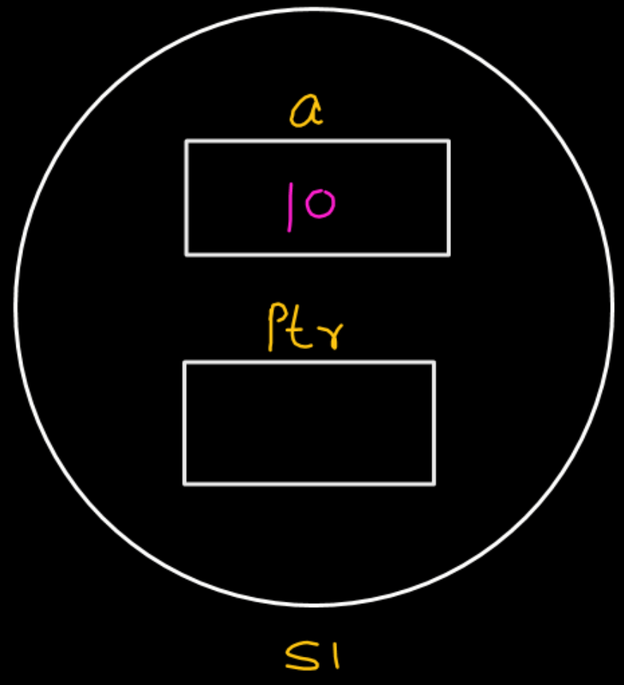
```
struct Pankaj{
    int a;
    float *ptr;
};
```

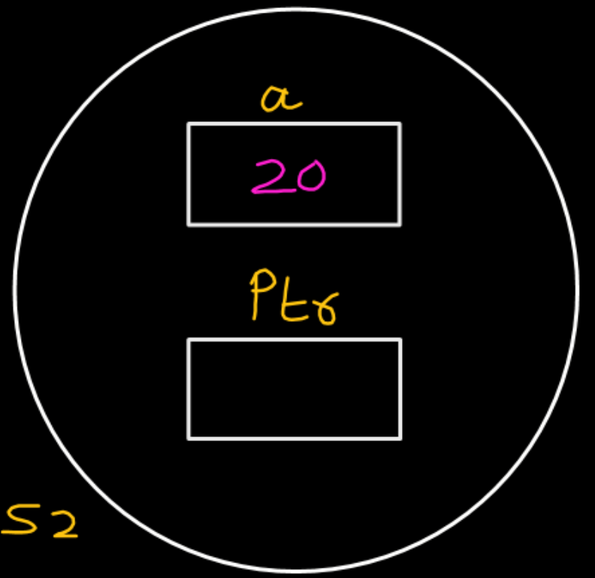
✓

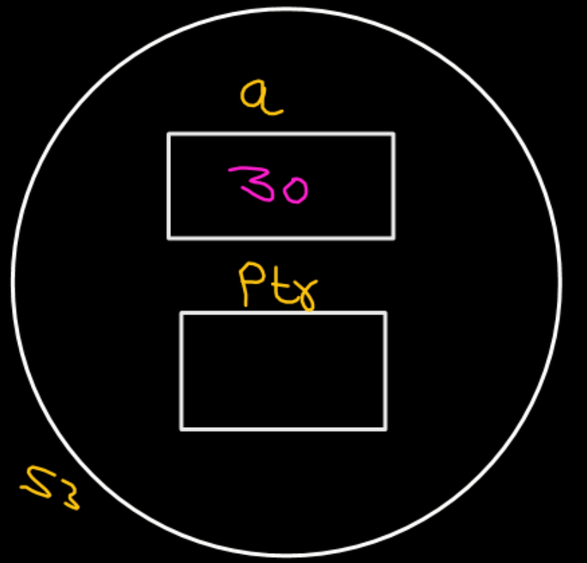
```
struct Pankaj{
    int a;
    struct Pankaj *ptr;
};
```

✓

```
struct Pankaj {
        int a;
struct Pankaj *Ptr;
        };
void main(){
   struct Pankaj S1,S2,S3;
    S1.a = 10;
    S2.a = 20;
    S3.a = 30;
    S1.Ptr = Address of
             struct Pankaj
             type
             variable
```

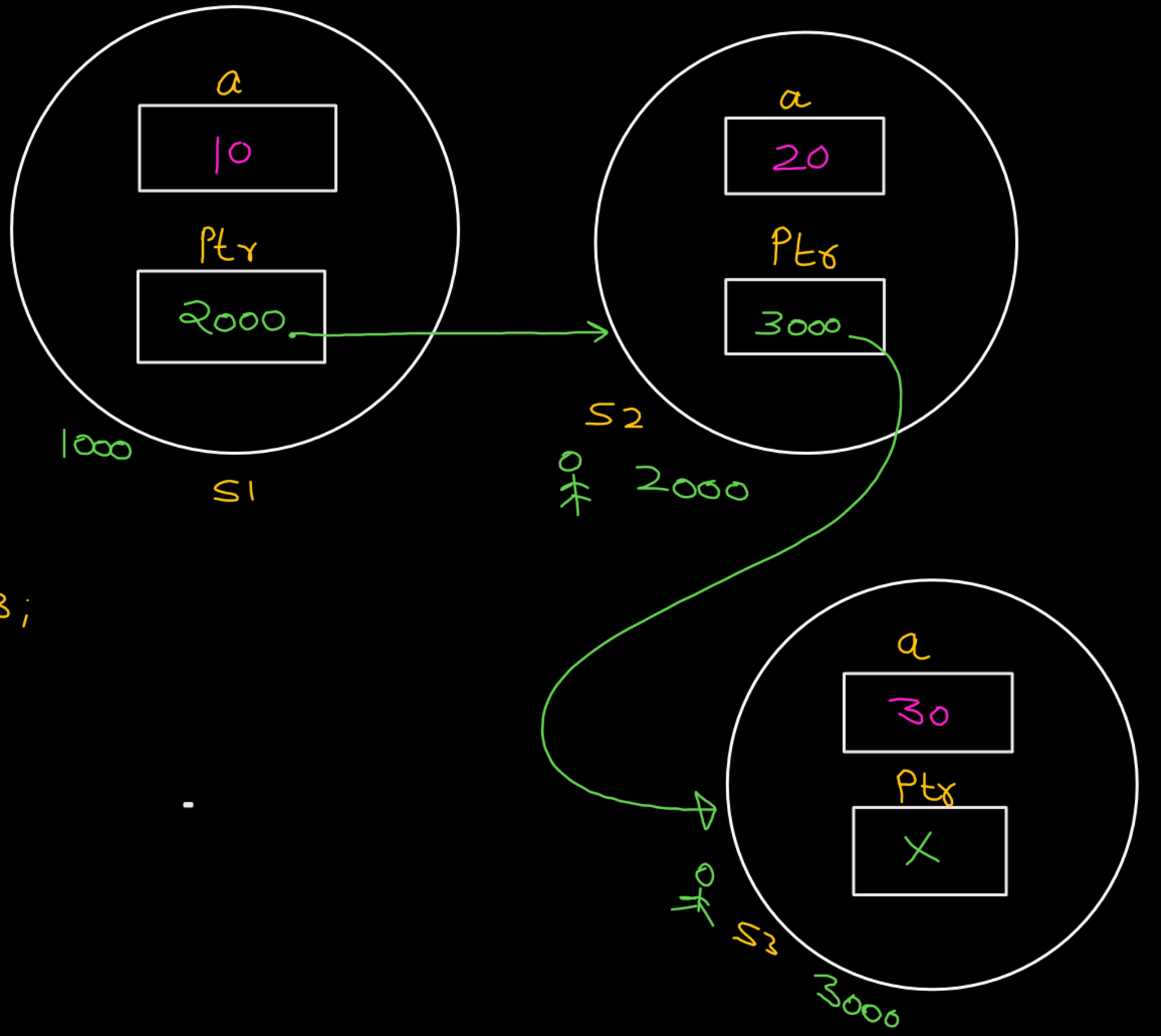






Self referential structure

```
struct Pankaj {
    int a;
    struct Pankaj *Ptr;
};

void main(){
    struct Pankaj S1,S2,S3;
    S1.a = 10;
    S2.a = 20;
    S3.a = 30;
    S1.Ptr = &S2;
    S2.Ptr = &S3;
    S3.Ptr = NULL;
}
```



Linked List

L.L is a linear data structure which is collection of elements called nodes, in which every node contain 2 parts

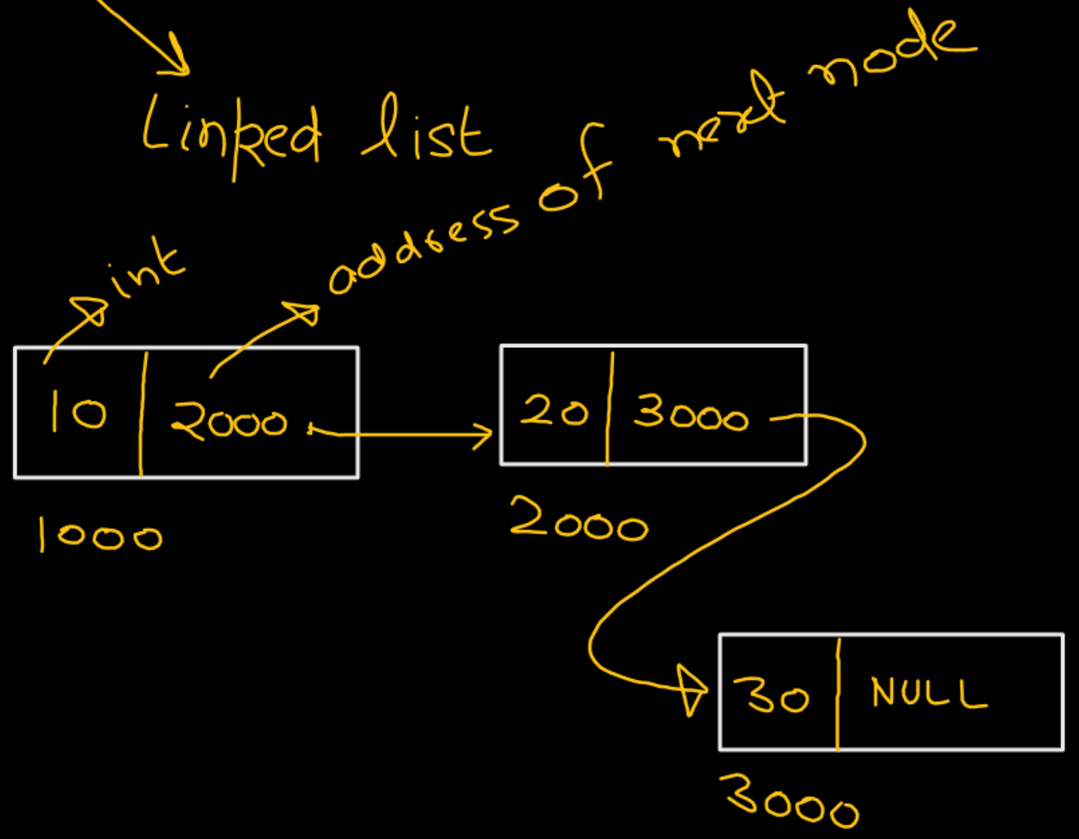
(i) data

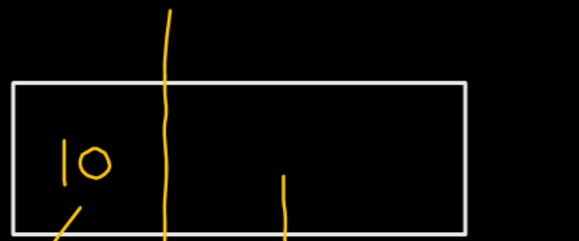
(ii) address of next node

Linear data structure

Array

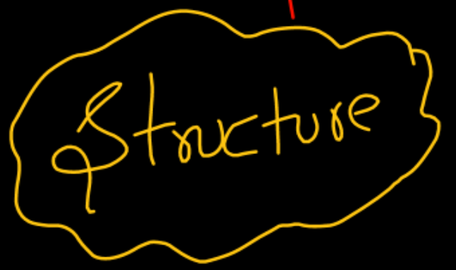
Linked list





Node is a collection of diff. types of elements

How to implement ?



```
struct Node{
        int data ;
  struct Node *next ;
            };

void main(){

      Insert(100);


}
```

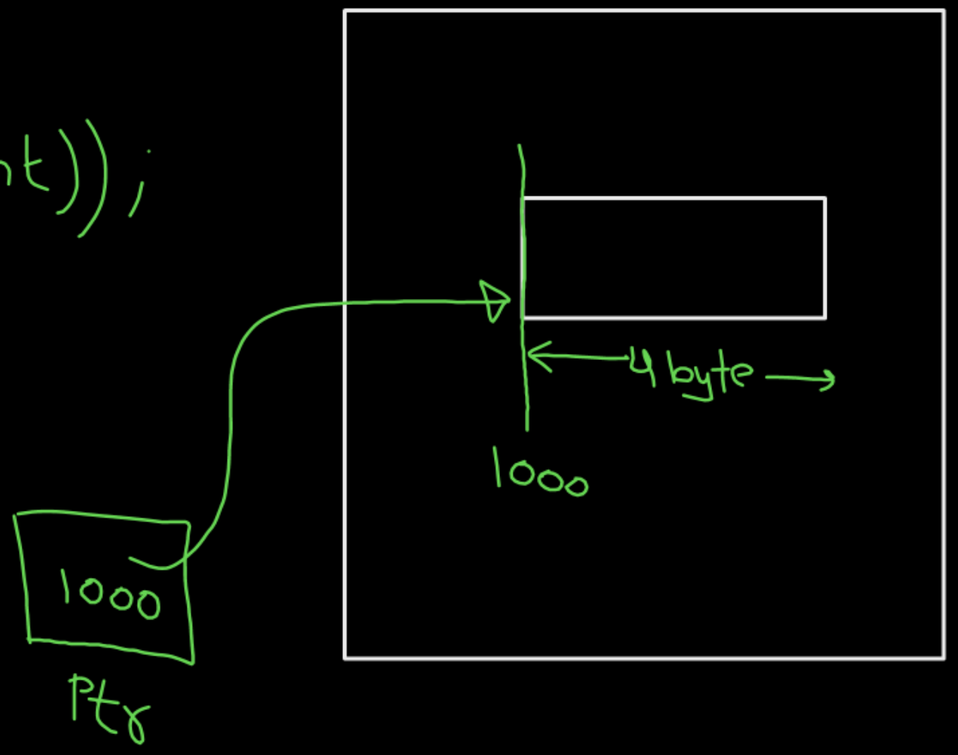
```
void Insert(int key){
  struct Node s1;
    s1.data = 10;
}
```

local var.

Node ⟶ malloc help

↳ through out program

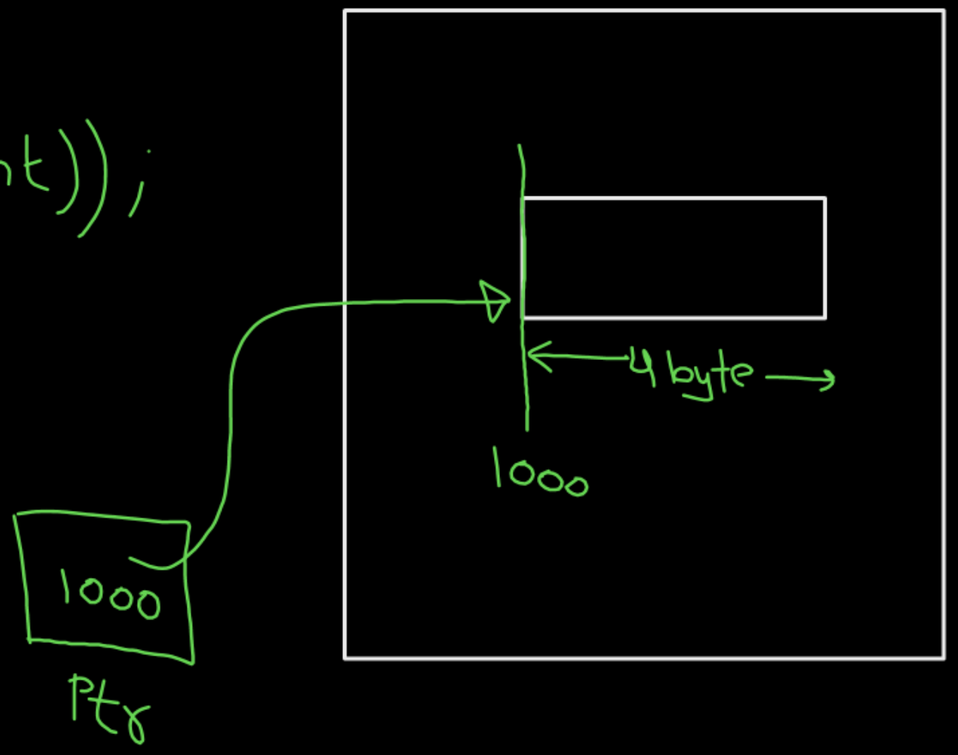
```
int *Ptr ;

Ptr = malloc( sizeof(int));
```



Node ⟶ malloc help

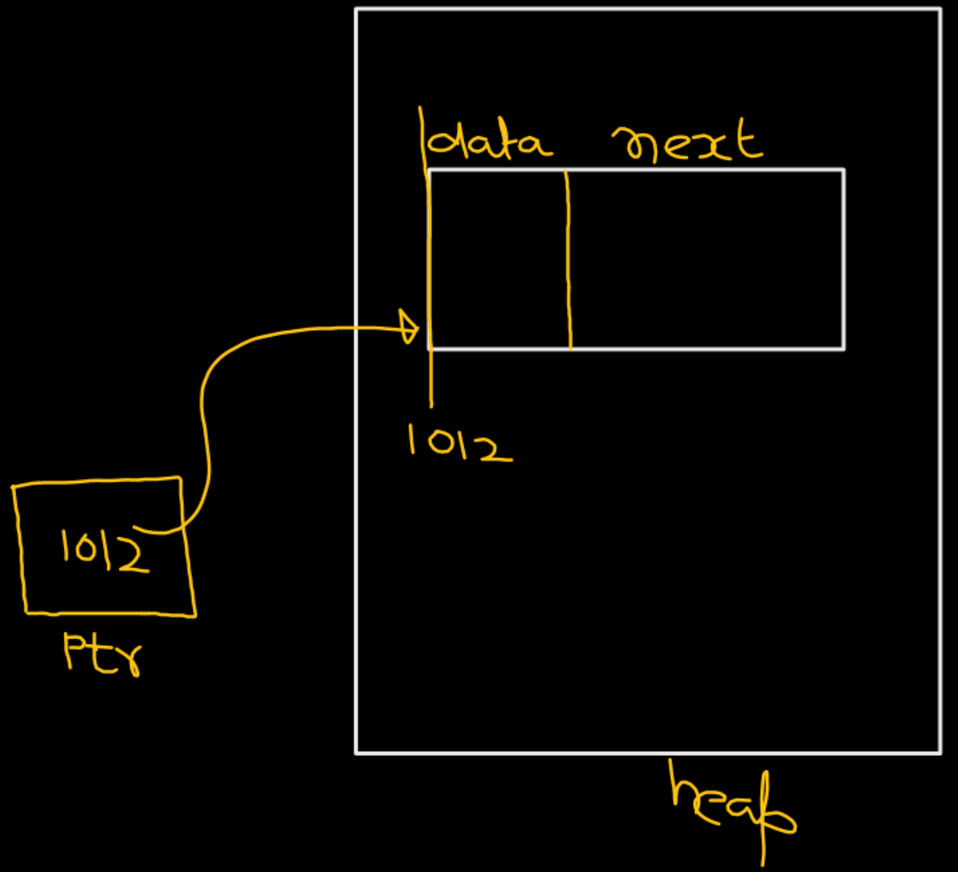
↳ through out program

```
int *Ptr ;
Ptr = malloc(sizeof(int));
float *Ptr ;
Ptr = malloc(sizeof(float));
```



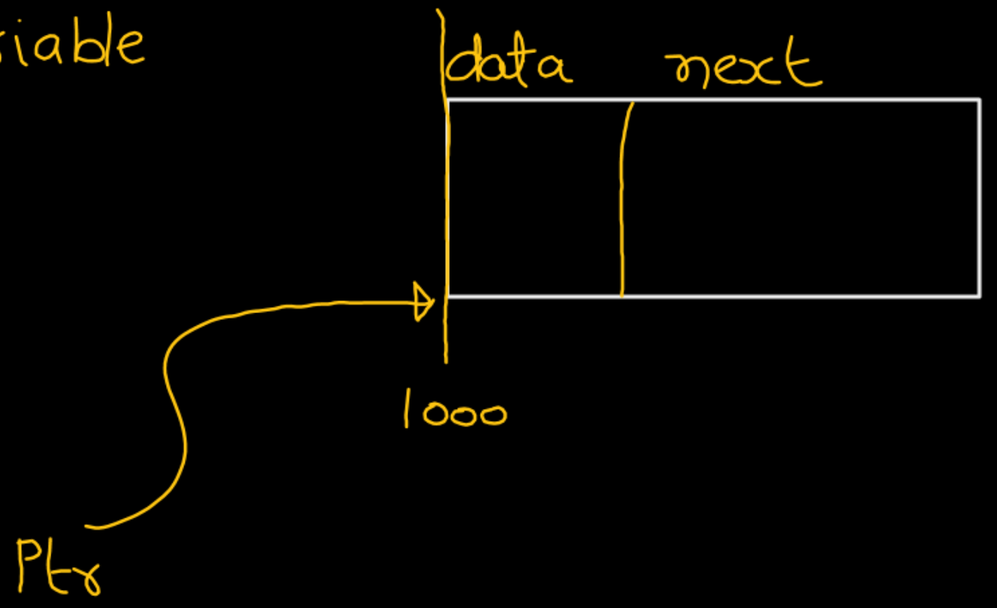
struct Node

```
struct Node *Ptr;
Ptr = malloc(sizeof(struct Node));
```



```
struct Node *Ptr ;
```

Ptr : Pointer to structure variable



How to access members from pointer to structure ?

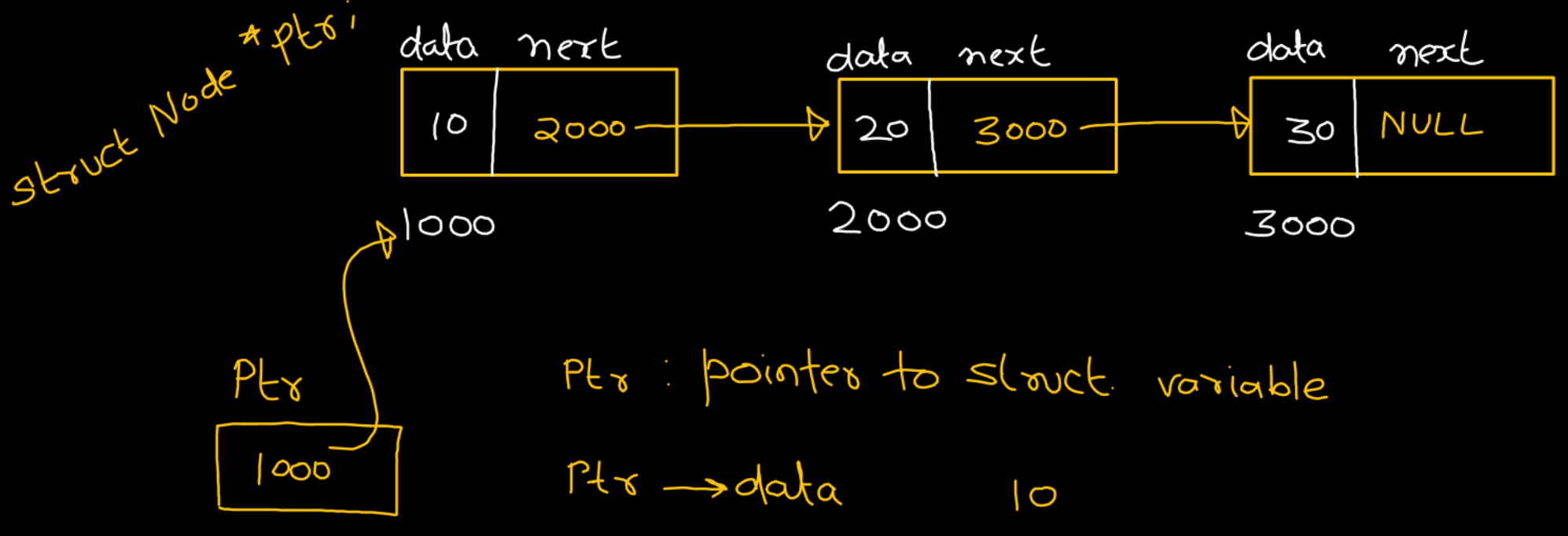
(i) `(*Ptr).data` OR `Ptr -> data`

(ii) `(*Ptr).next` OR `Ptr -> next`

```
struct Node{
    int data;
struct Node *link;
    };
```

OR

```
struct Node{
    int data;
  struct Node *next;
    };
```



Ptr : pointer to struct variable

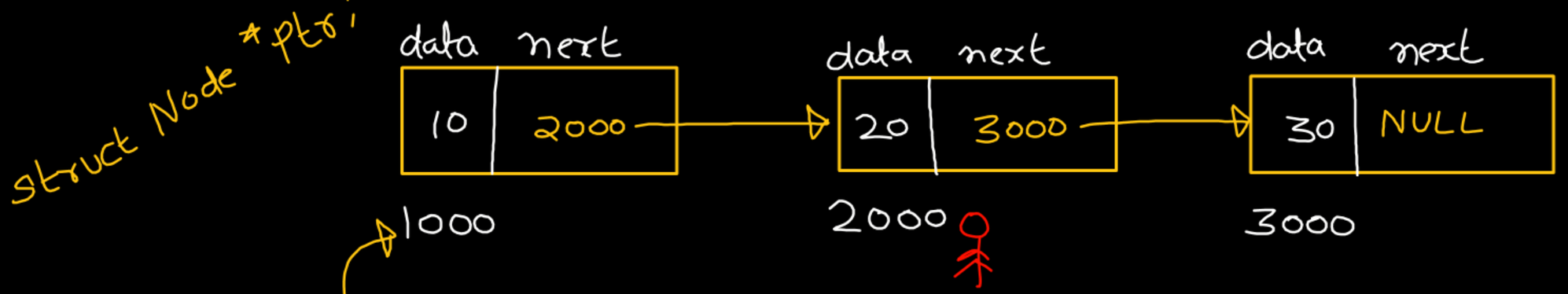
Ptr → data 10

Ptr → next : 2000 (Address of second node)

It is also pointer to a node

Pointer to second node

Pointer to struct variable



Ptr : pointer to struct variable

Ptr → data : 10

Ptr → next : 2000 (Address of second node)

It is also pointer to a node

Pointer to second node

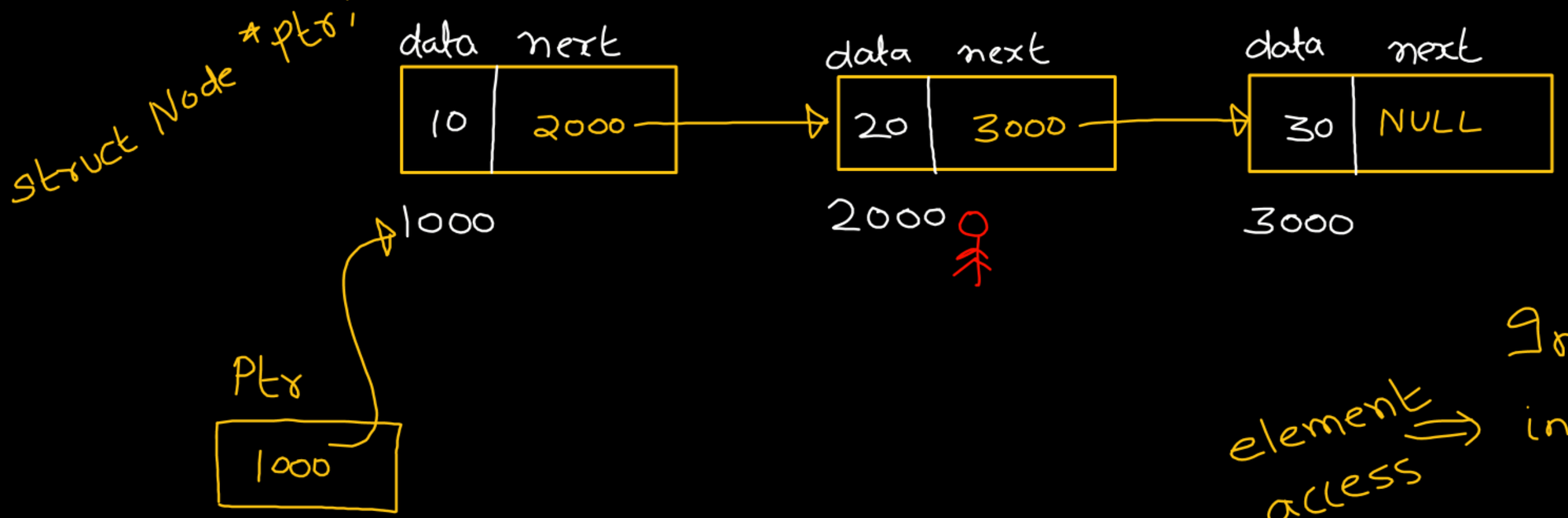
Pointer to struct variable

Pointer to 2nd node

(Ptr → next) → data : 20

(Ptr → next) → next

2nd node का next : 3rd node का address field



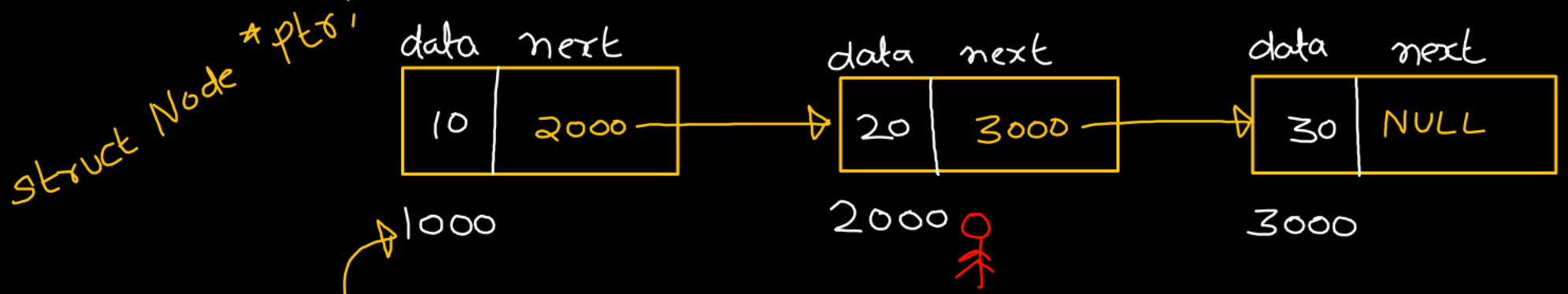
In array

element ⟹ index

access

Next element

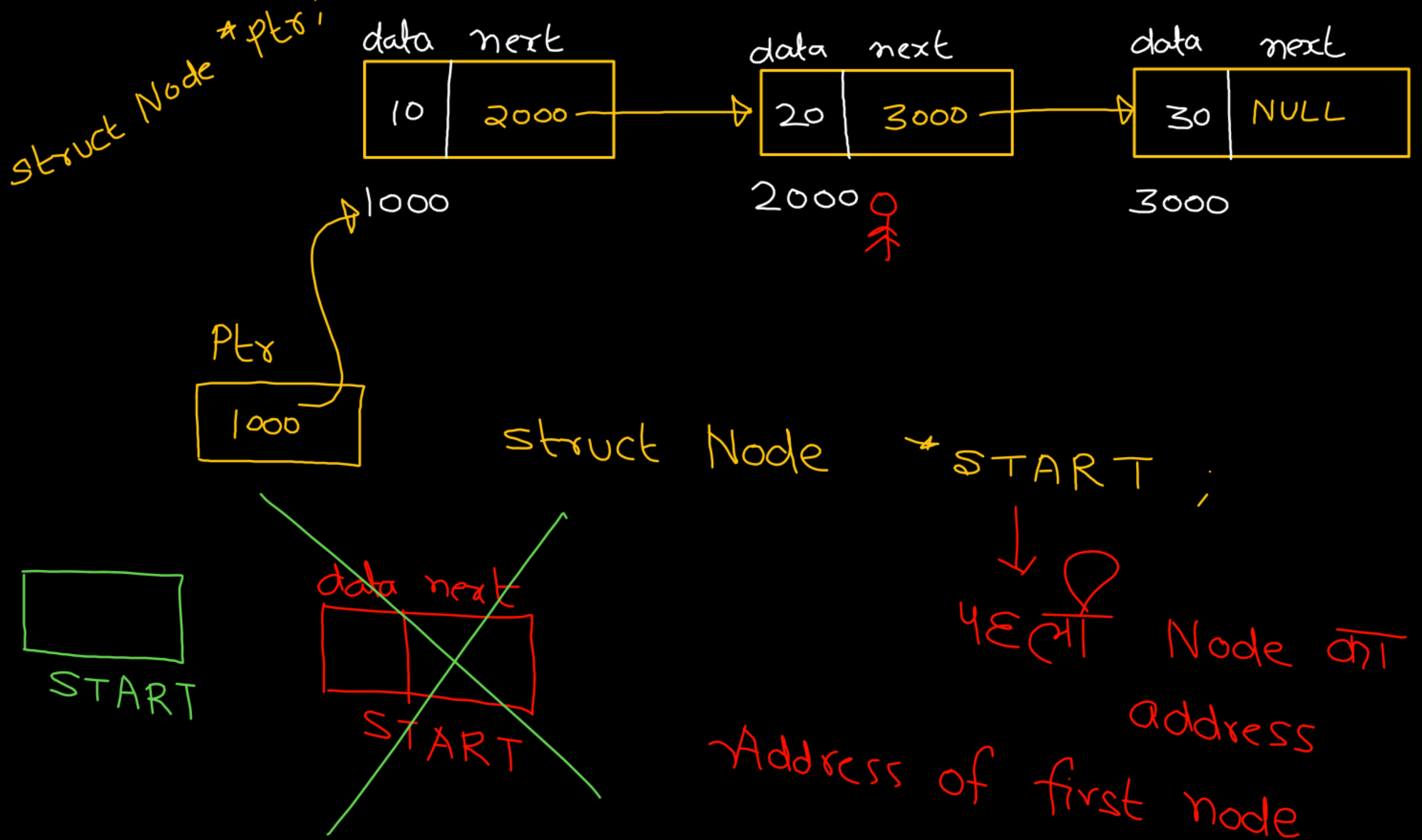
⟹ index++

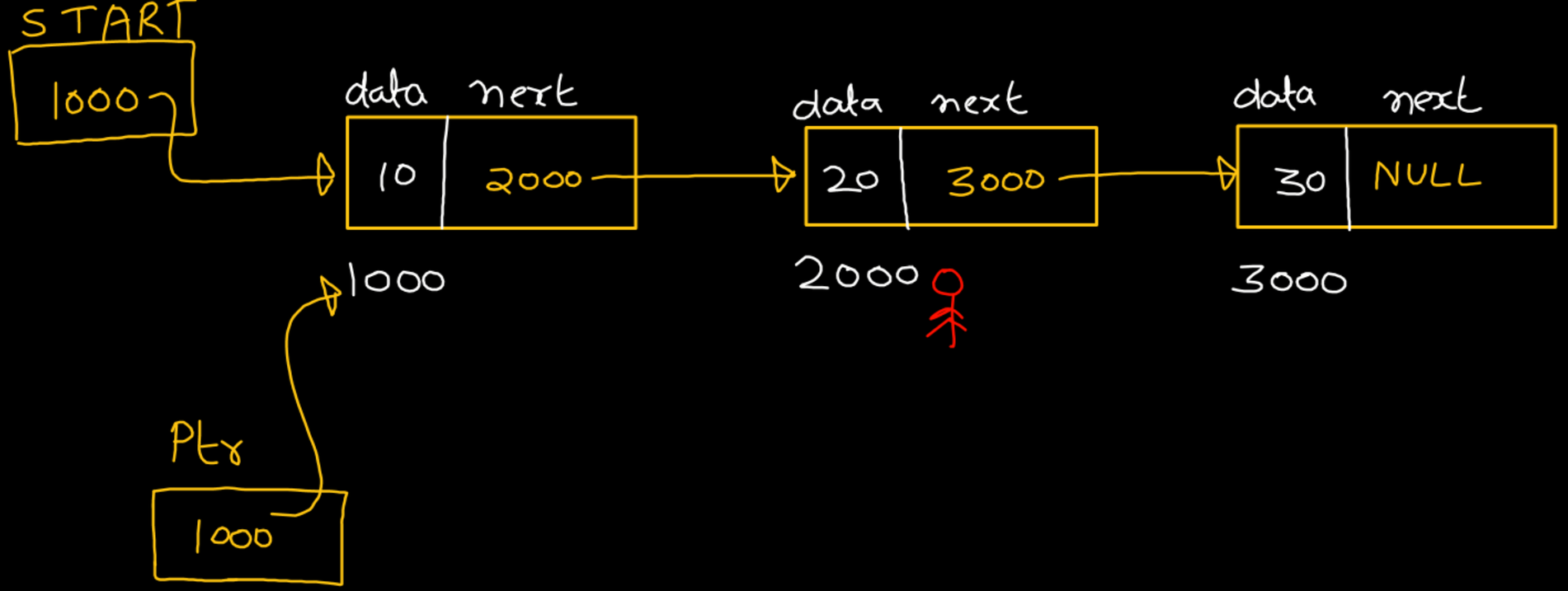


How to access next node?

(2000)

Ptr = Ptr → next


struct Node *ptr;
data next
10 2000
data next
20 3000
data next
30 NULL
1000
2000
3000
Ptr
1000
struct Node *START;
START
data next
START
पहले Node का address
Address of first node


START
1000
data
next
10
2000
data
next
20
3000
data
next
30
NULL
1000
2000
3000
Ptr
1000

Empty L.L

⇒ there is no 1st node

⇒ START can not have any valid address

i.e START = NULL

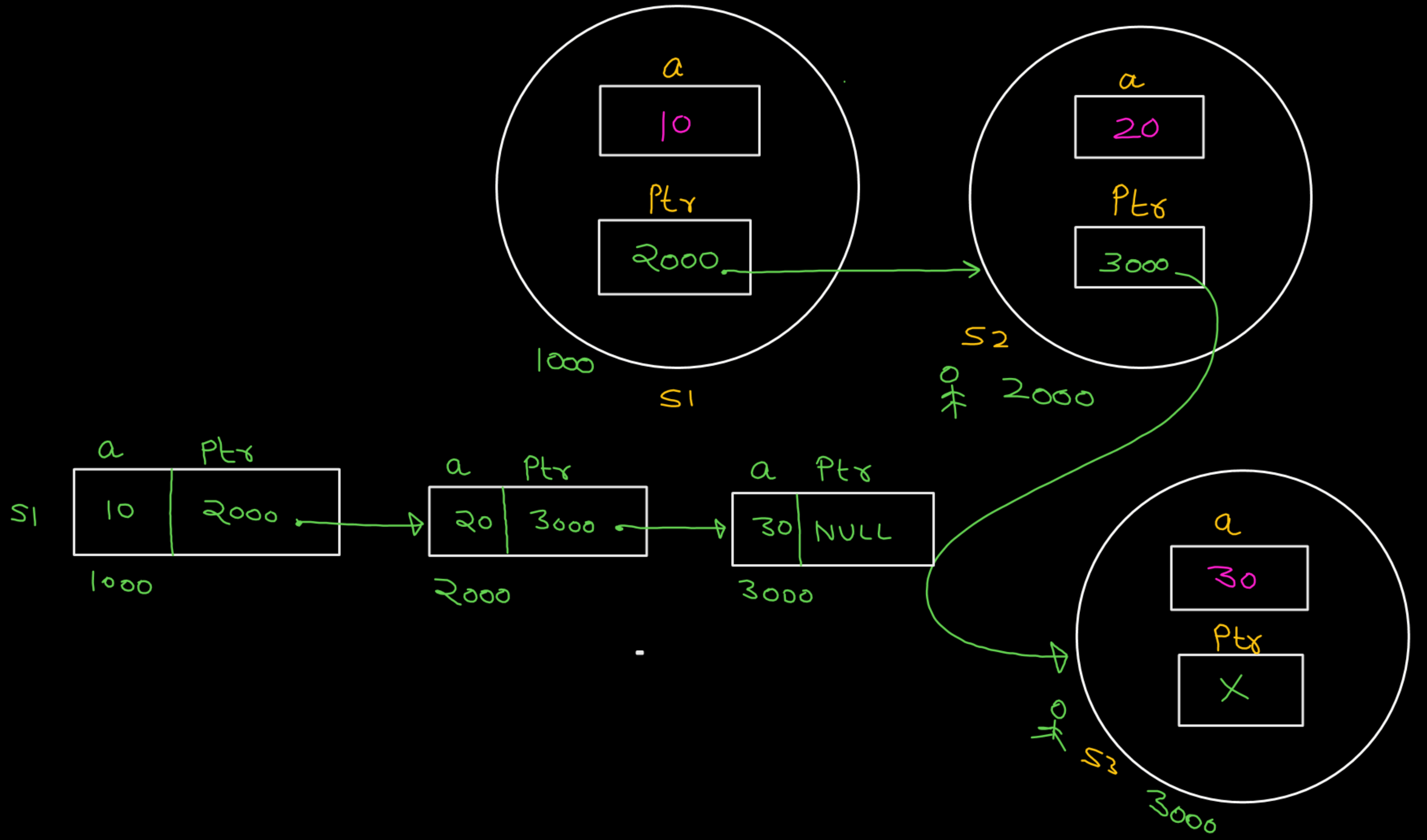
```
struct Node {
        int data ;
struct Node *Link ;
        }*START = NULL;
```

NULL ⟶ data

NULL ⟶ next

Error

Doubt ?


a
10
Ptr
2000
1000
S1
a
20
Ptr
3000
S2
2000
a
Ptr
10
2000
S1
1000
a
Ptr
20
3000
2000
a
Ptr
30
NULL
3000
a
30
Ptr
X
S3
3000

```
int a = 10;
int b = 20;
int c = 30;
int *Ptr;
Ptr = &a;
Ptr = &b;
Ptr = &c;
```


THANK YOU GW
SOLDIERS !