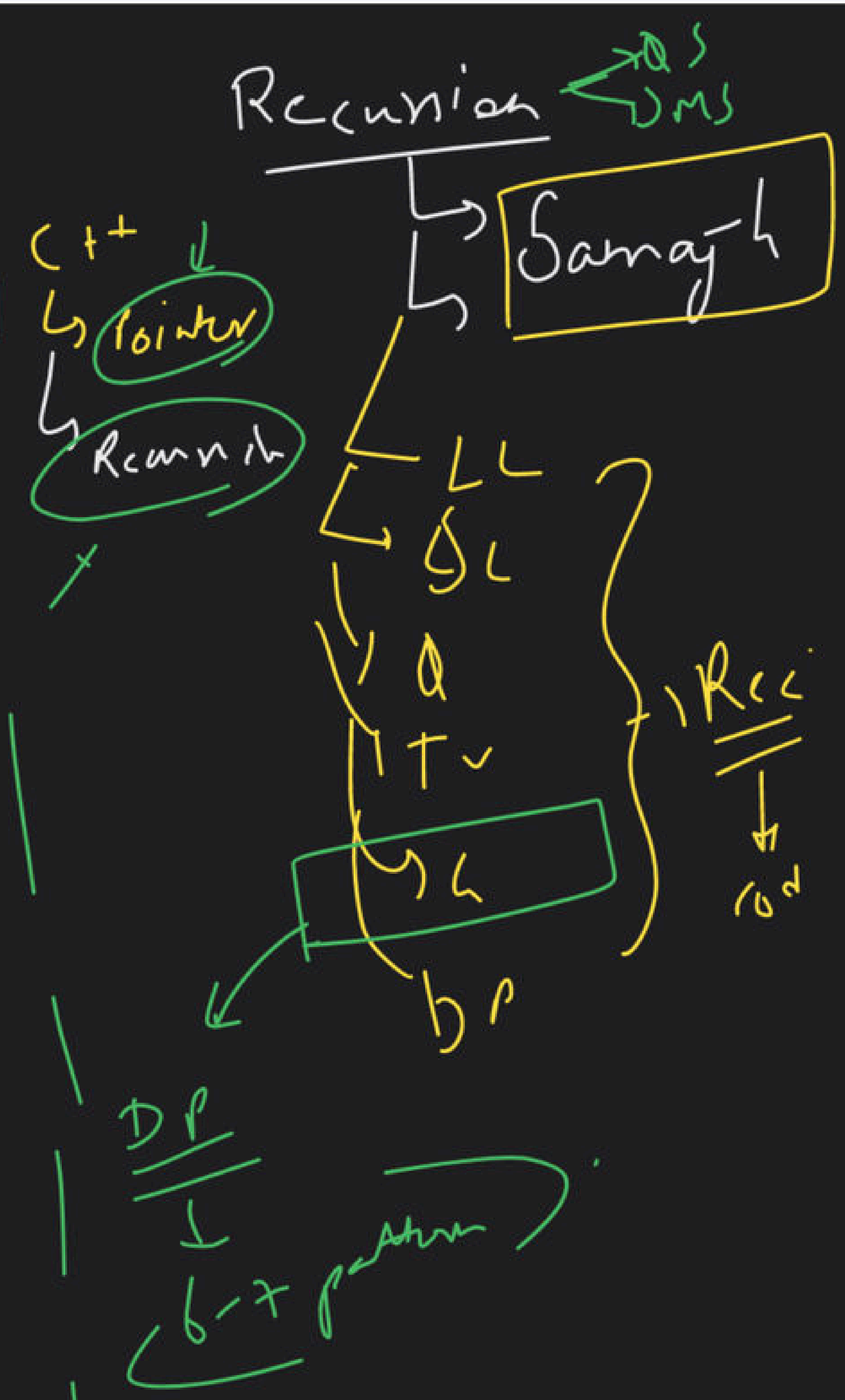
Doubt Clearing Session - Part VII

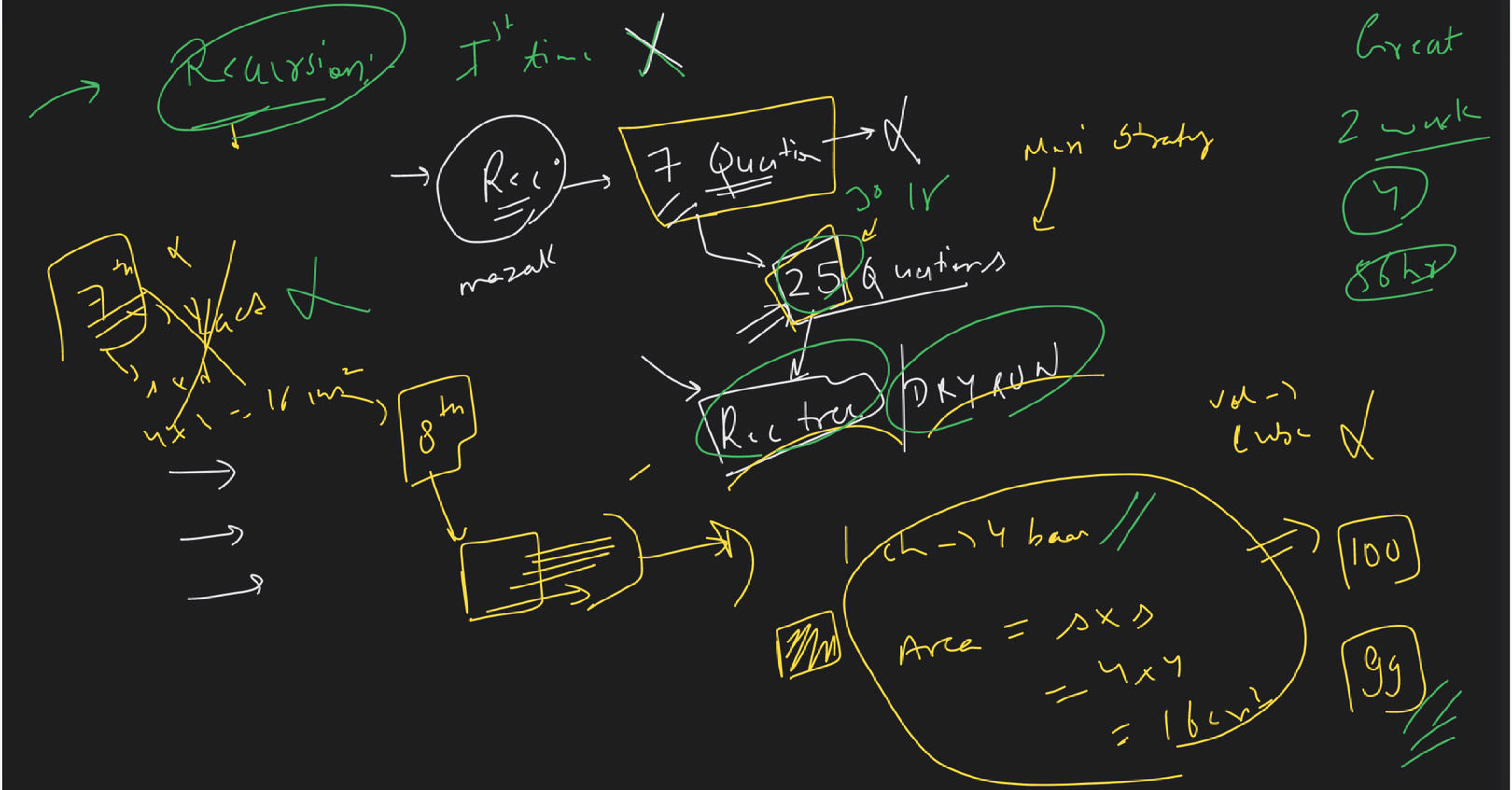
Foundation Course on Data Structures & Algorithm - III

Love Babbar • Lesson 7 • May 29, 2022

Doubt Session - VII

Sheet





Recursion

1st time ✗

Great

2 work

4

56 hr

Rec

mazak

7 Question → ✗

30 hr

25 Questions

Mani Strategy

Rec tree

DRY RUN

7th

Maths ✗

4×4 = 16 cm²

8th

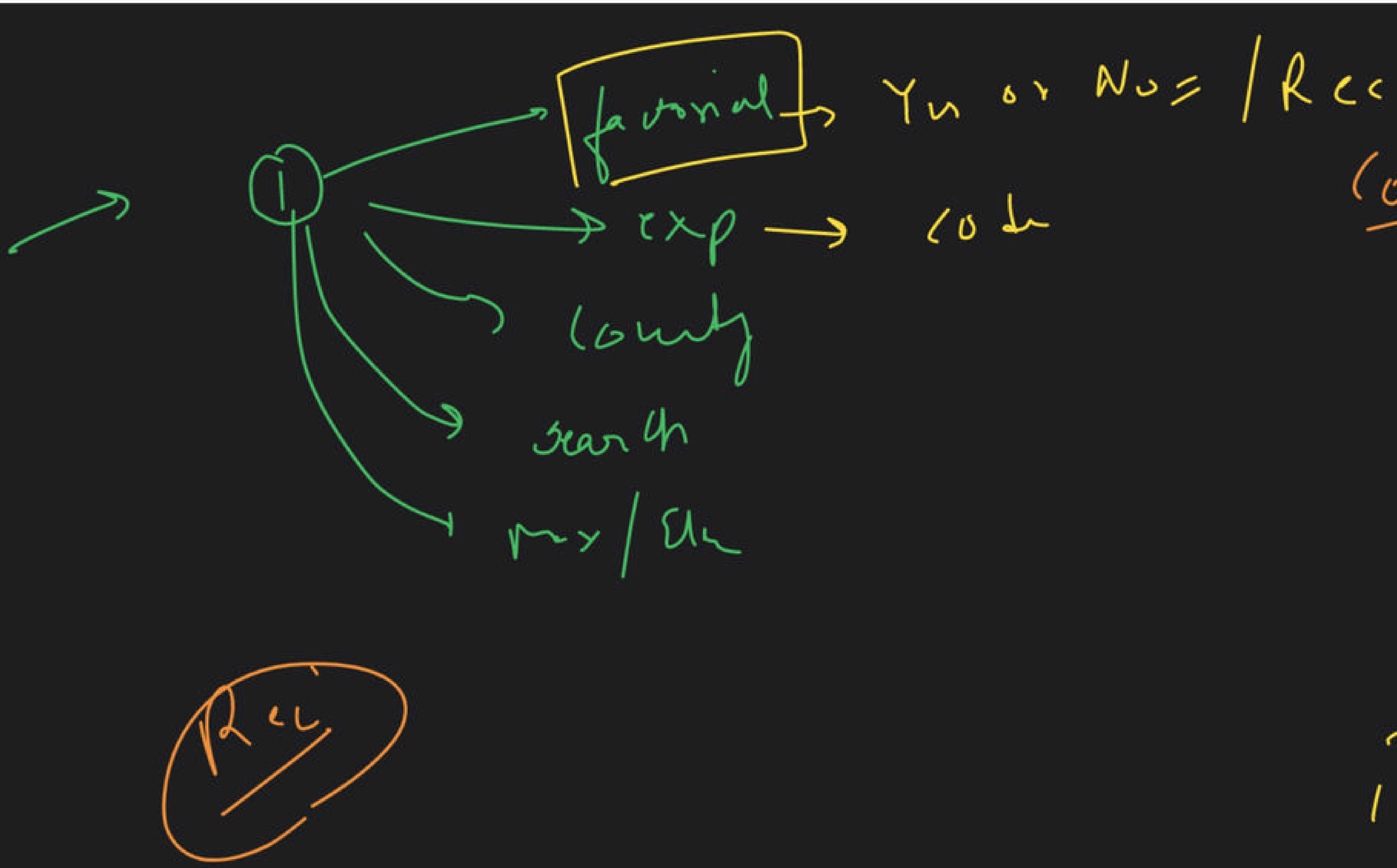
vol → cube ✗

1 ch → 4 baar

Area = s×s
= 4×4
= 16cm²

100

99

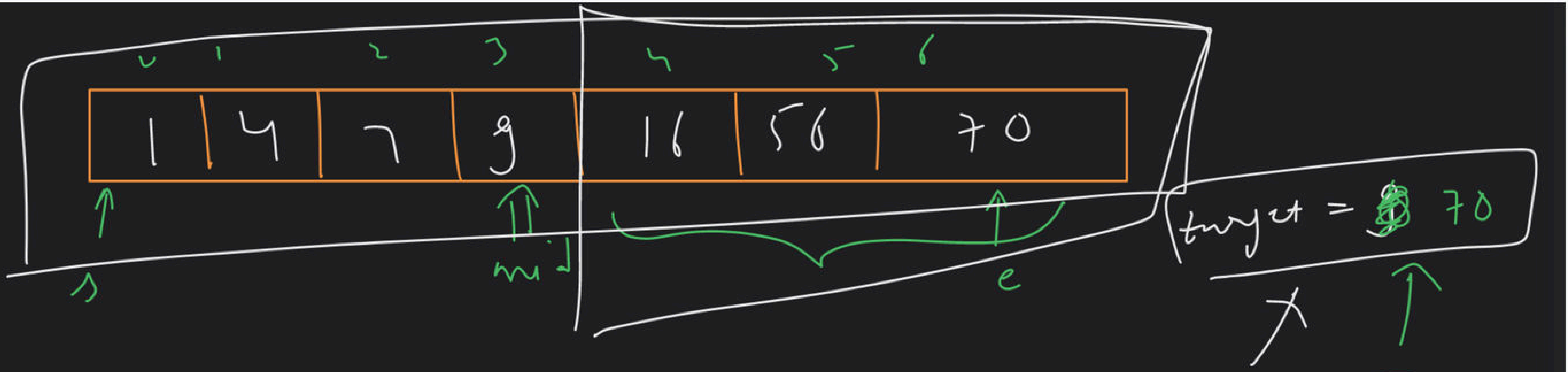


Code :-

```
if (n == 0)
    return 1;

int ans = fun (2, n/2)

if (n&1)
    return ans*ans×2;

else
    return ans*ans;
```

Binary search

```
int  BS( int arr[], int n, int target)
{
     int s = 0
     int e = n-1;
     int mid = (s+e)/2   or   s + (e-s)/2
     while (s <= e)
     {
          if (arr(mid) == target)
               return mid;
          if (arr(mid] < target)
               s = mid+1;
          else
               e = mid-1
          mid = (s+e)/2;
     }
     return -1;
```

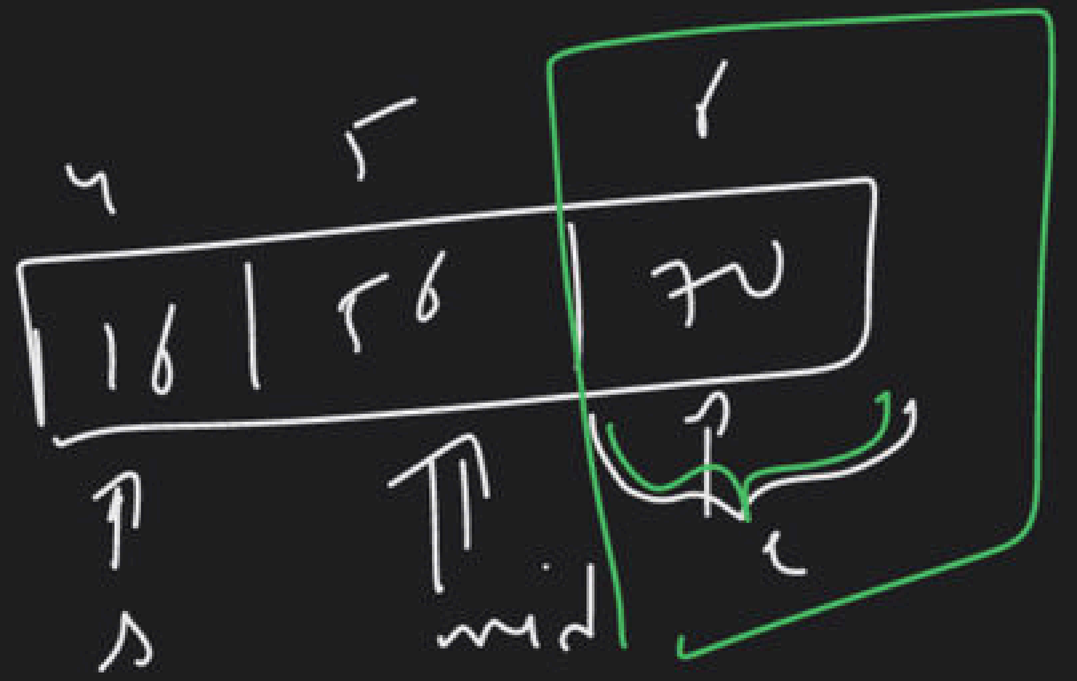


$mid = \frac{0+6}{2} = 3$

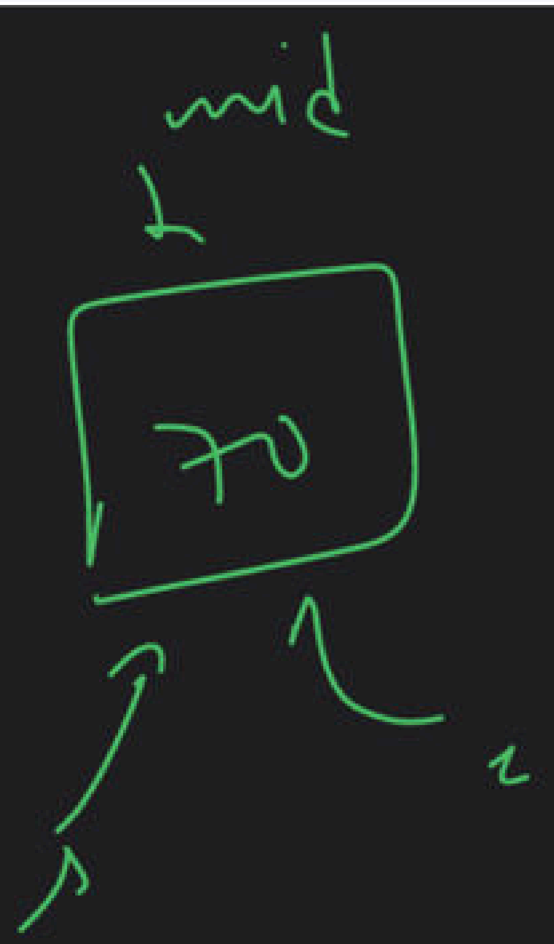
arr[mid] == ~~7~~70 → false

arr[mid] < 70 → true →

9

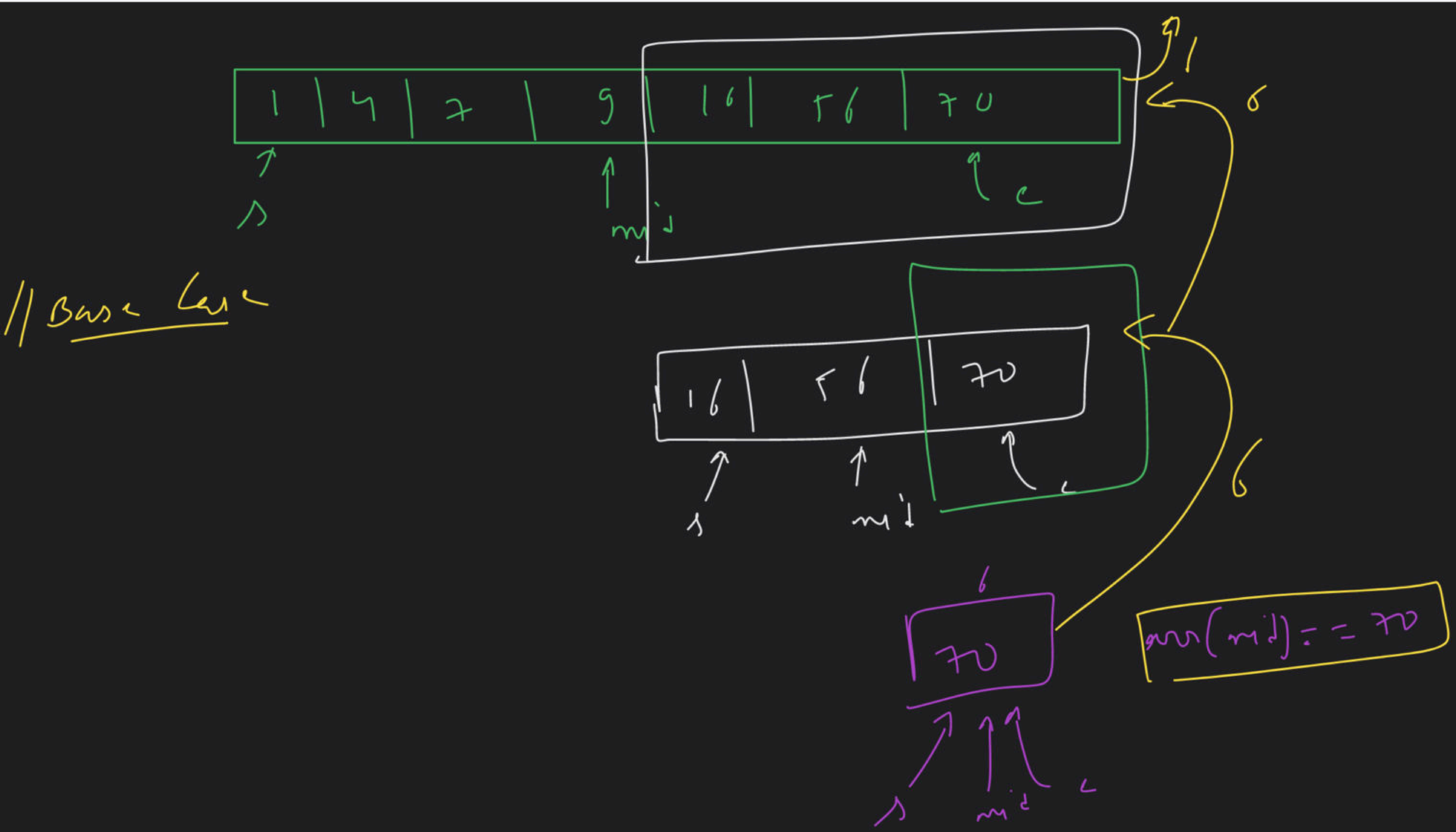


arr(mid) < 70



arr(mid) == 70

return mid



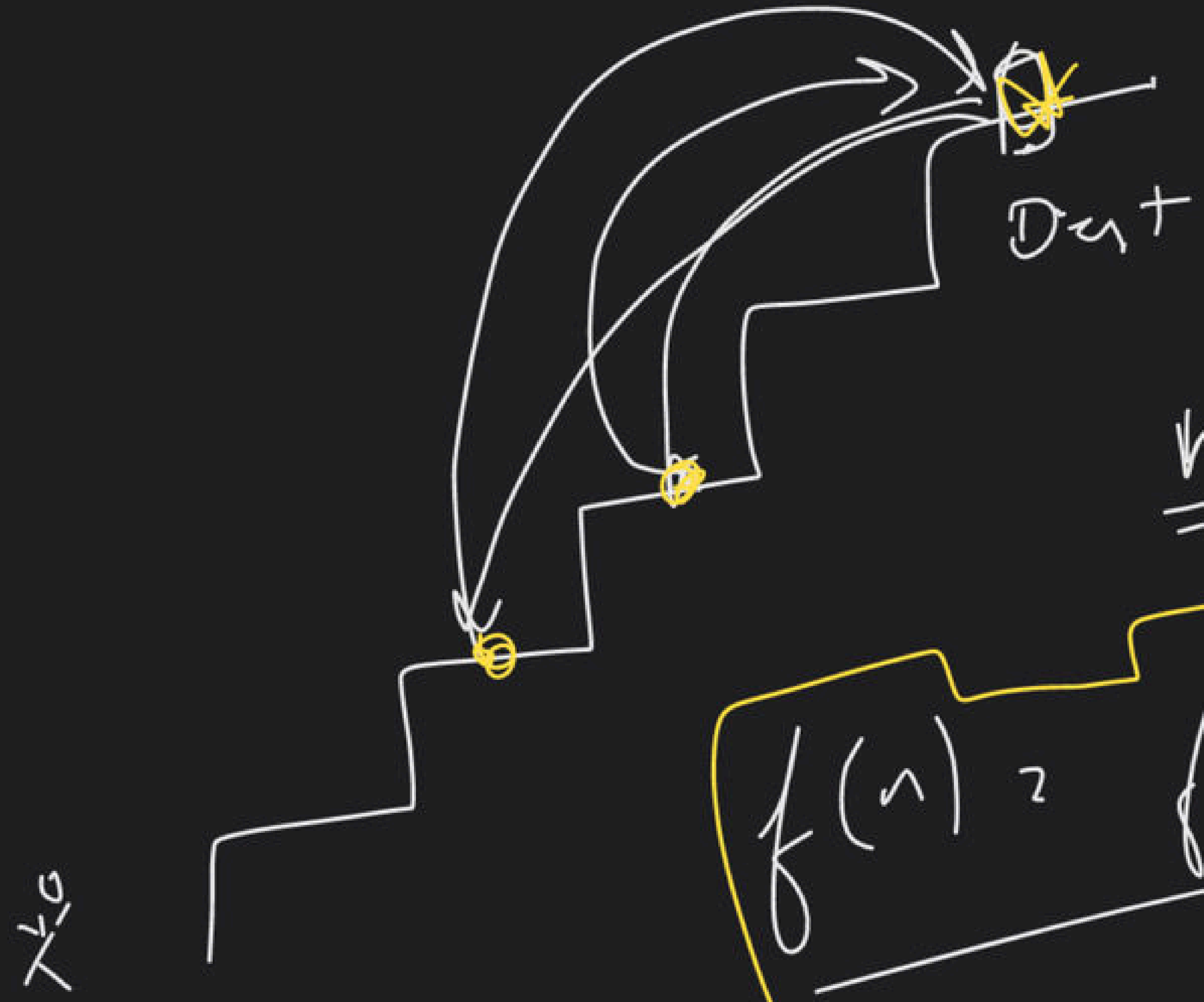
// Base Case

arr(mid) == 70

```
func ( arr , s , e , target)

} // B.C

Not found
    if( s > e) → entire array trav
        return -1;

Found    int mid = $\frac{(s+e)}{2}$ ;

    if ( arr[mid] == target)
        return mid
                                                s = mid+1

if ( arr (mid] < target) → right part
    return func ( arr , mid+1 , e , target);    e = mid-1
    else → left part
        return func(arr, s, mid-1, target)
```

fibonacci

$\boxed{2, 3}$



n^{th}

$$\boxed{f(n) = f(n-2) + f(n-3)}$$

$\boxed{25}$

10h 4

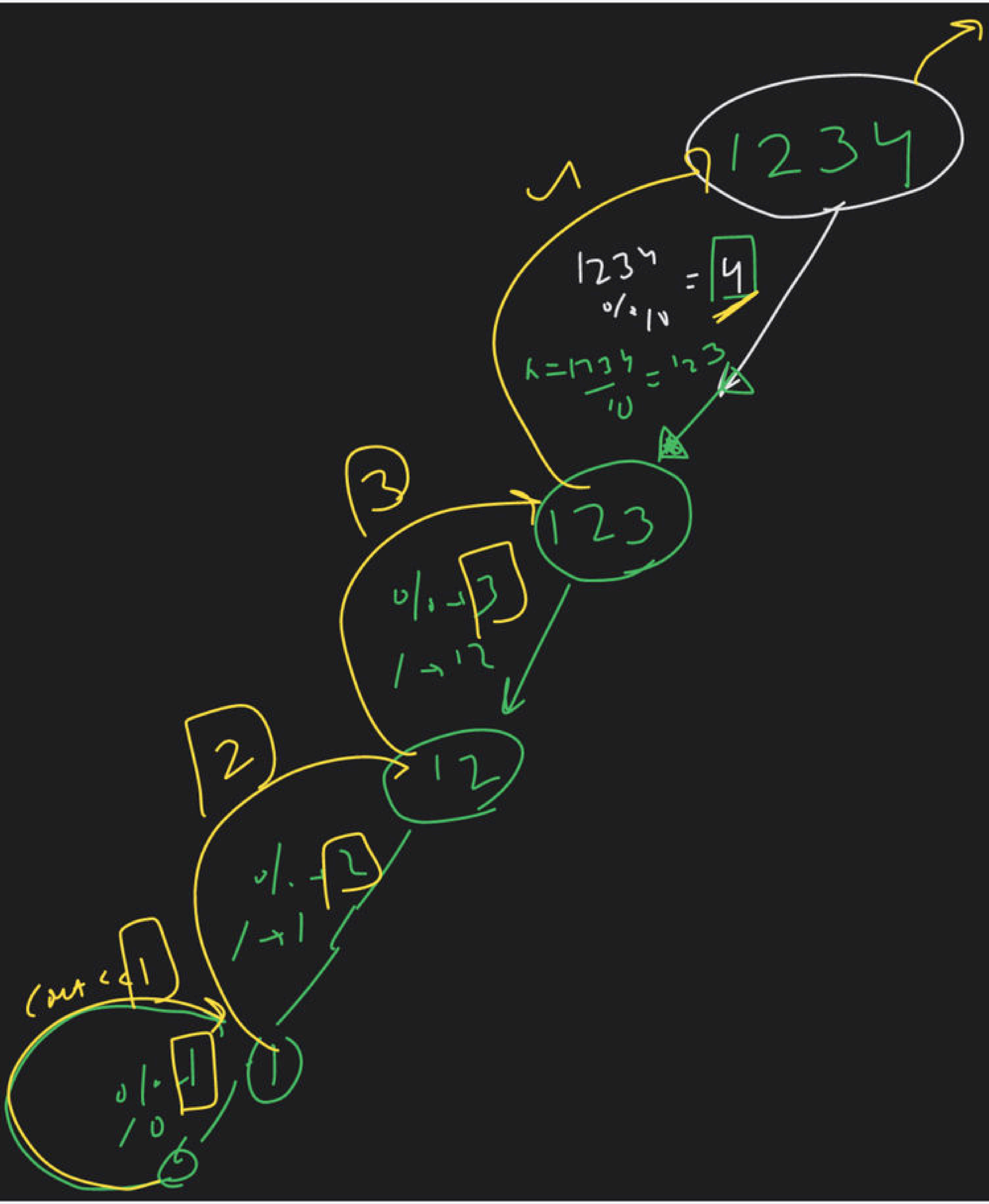
Recursion

① $\boxed{n = 1234}$

print its digits [Using Recursion]

1, 2, 3, 4

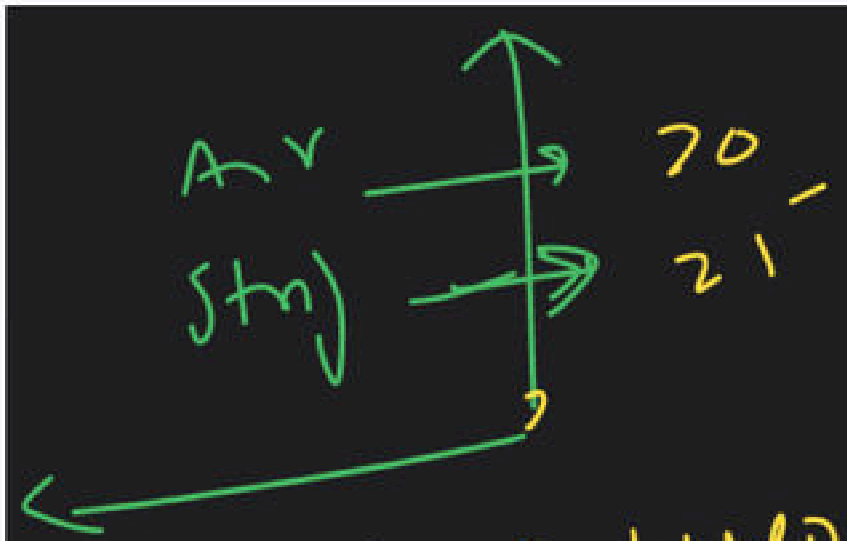
1234



B.C

if(n==0)

return;



LC ⇒ HARD X

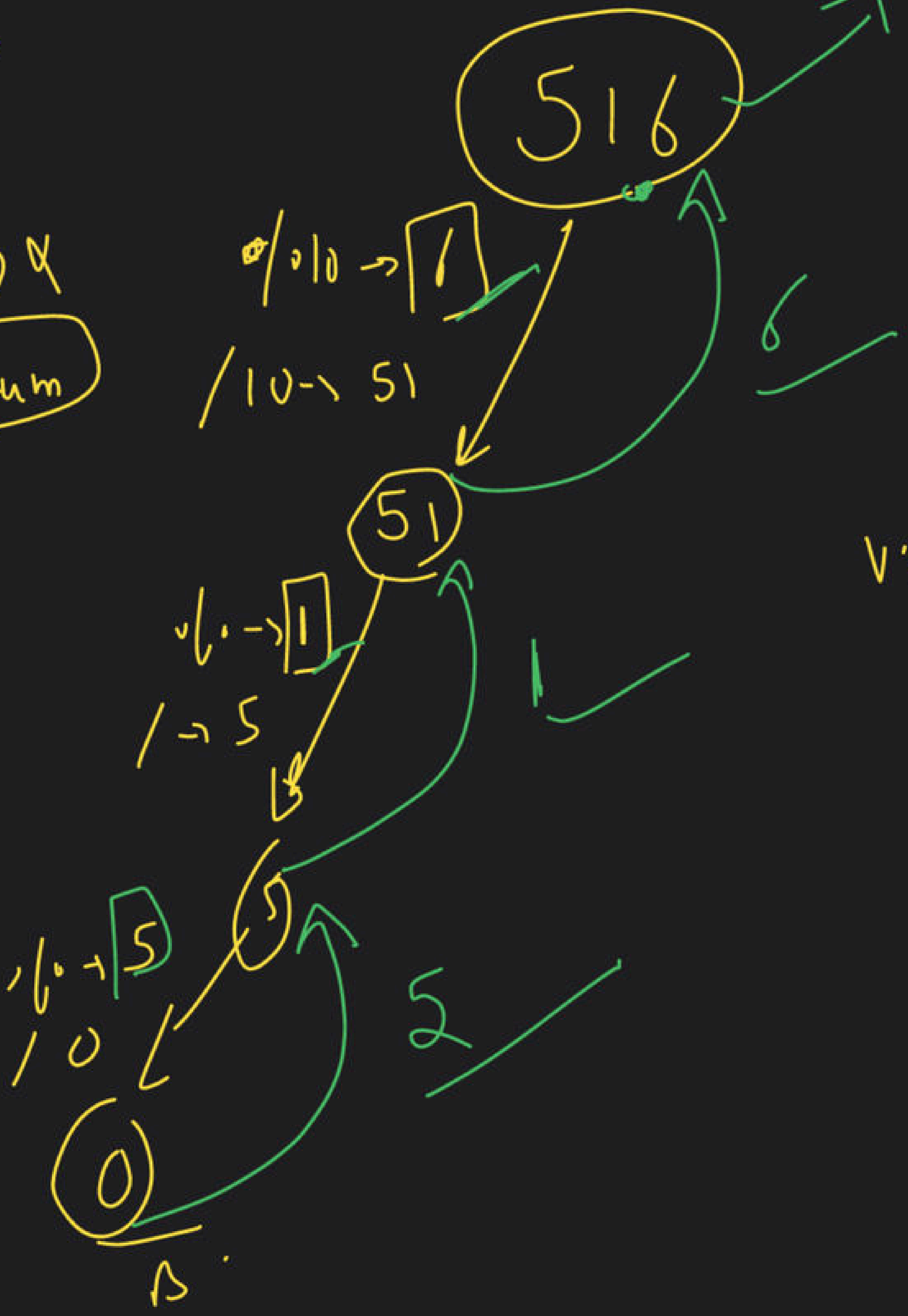
→ Medium

→ Easy

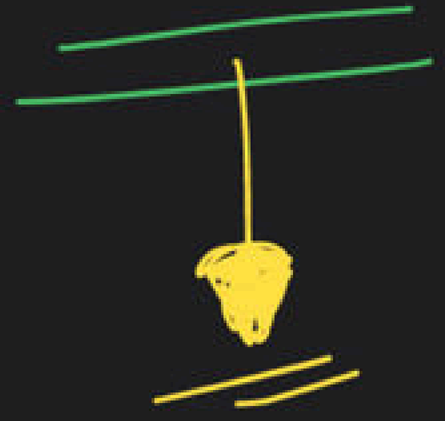
what ?

Discuss

DRY RUN



Code:



void

516

5, 1, 6

Ques

```
void   printDigits ( int n)
{    // Base case
          if ( n == 0)
                return;

     int  digit = n%10;
     n = n/10;
     printDigits (n);
     cout << digits << "_";
}
```

516

5, 1, 6

Yes or No

5 1 6

Five One six

$n = \boxed{516}$

o/p $\boxed{615}$

using rec.

n

5 1 6

5, 1, 6

digit

↳ num

516

$\rightarrow$ 516

% $\rightarrow$ 6

sum = 0

sum = sum * 10 + digit

= 0 × 10 + 6

= $\boxed{6}$

$\frac{+11}{10}$, 51

$\rightarrow$ 51 % 10 = $\boxed{1}$

sum = 6 × 10 + 1

= $\boxed{61}$

51/10 = 5

$\rightarrow$ 5 % 10 = $\boxed{5}$

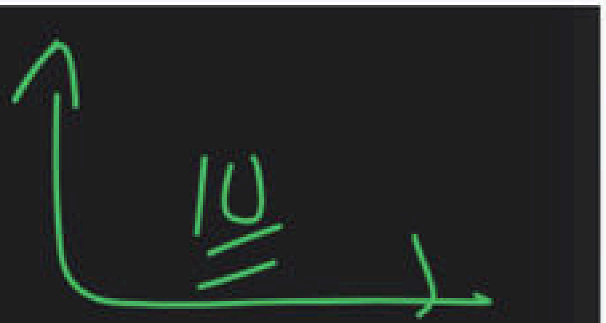
sum = 61 × 10 + 5

= 611

5/10 = $\boxed{0}$ $\rightarrow$ return

```
void reverse ( int n , int ~~&~~sum)
{
    // B.C
    if(n==0)
    { cout << sum;
      return;
    }
    int digit = n%10;
    sum = sum*10 + digit;
    n = n/10    i.v
    reverse(n, sum);
}
```

R.C



Gadbad

→ naam

→ (511 , 0)

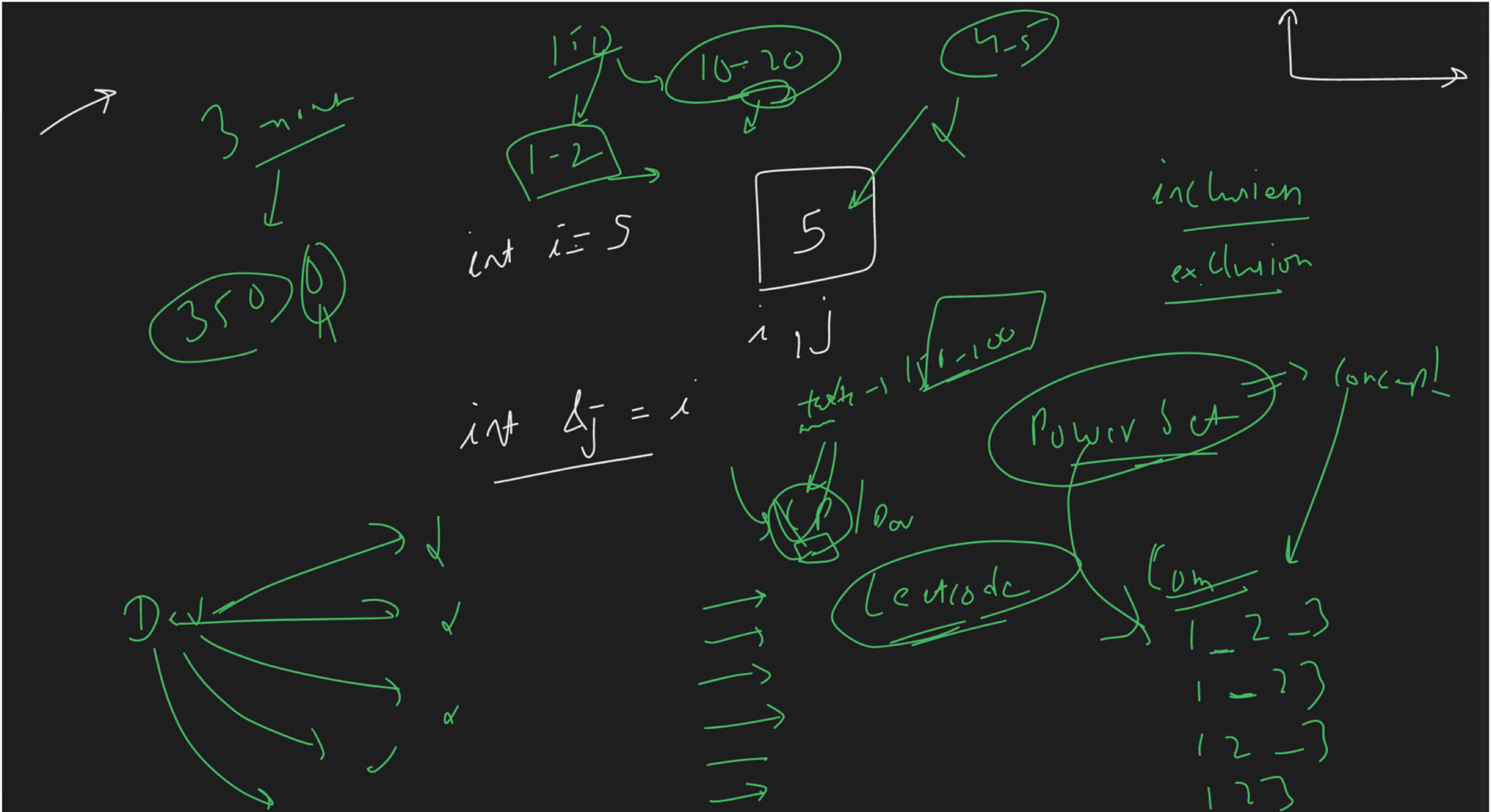
dam -> badla

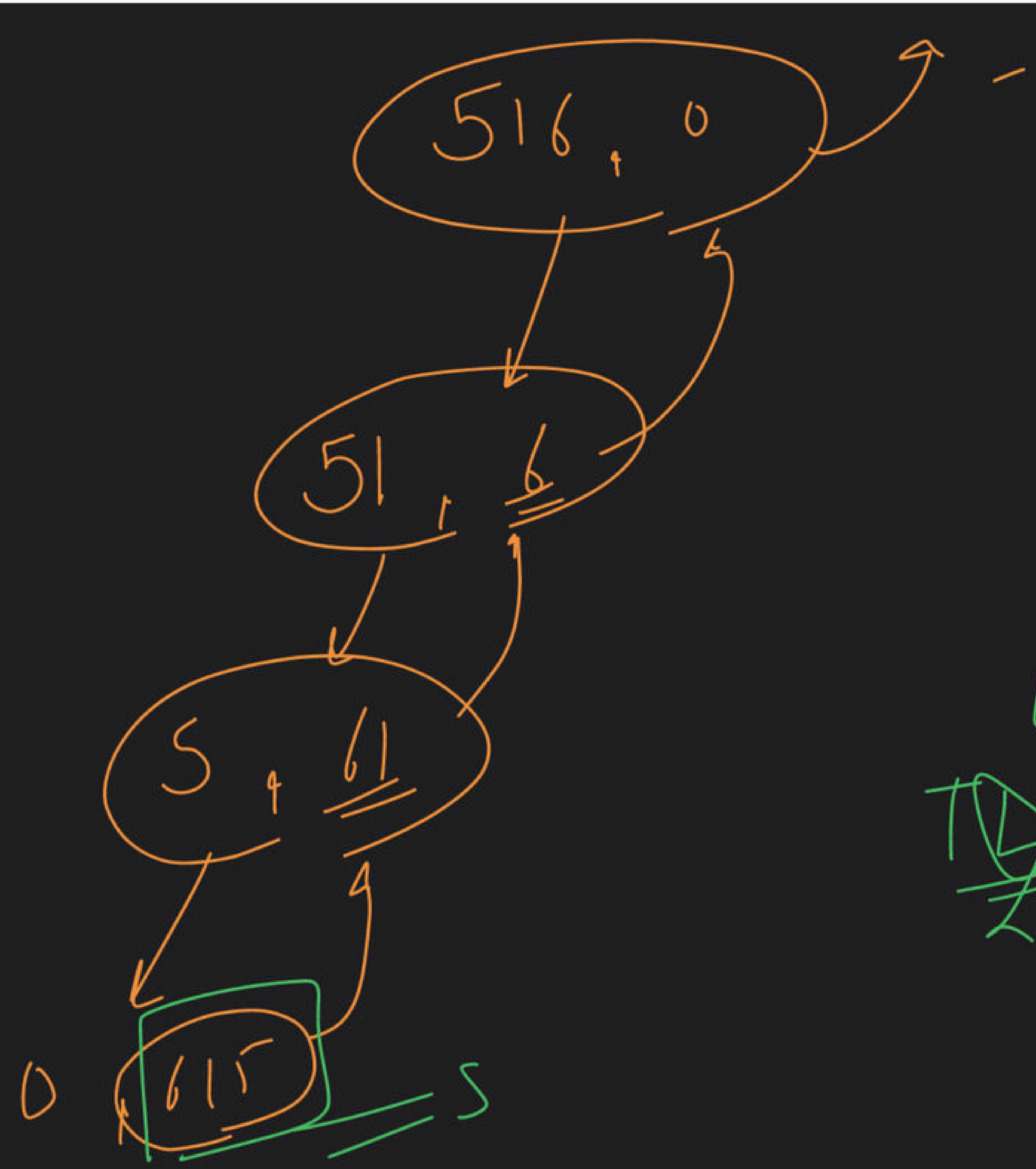
reverse(516, 0)

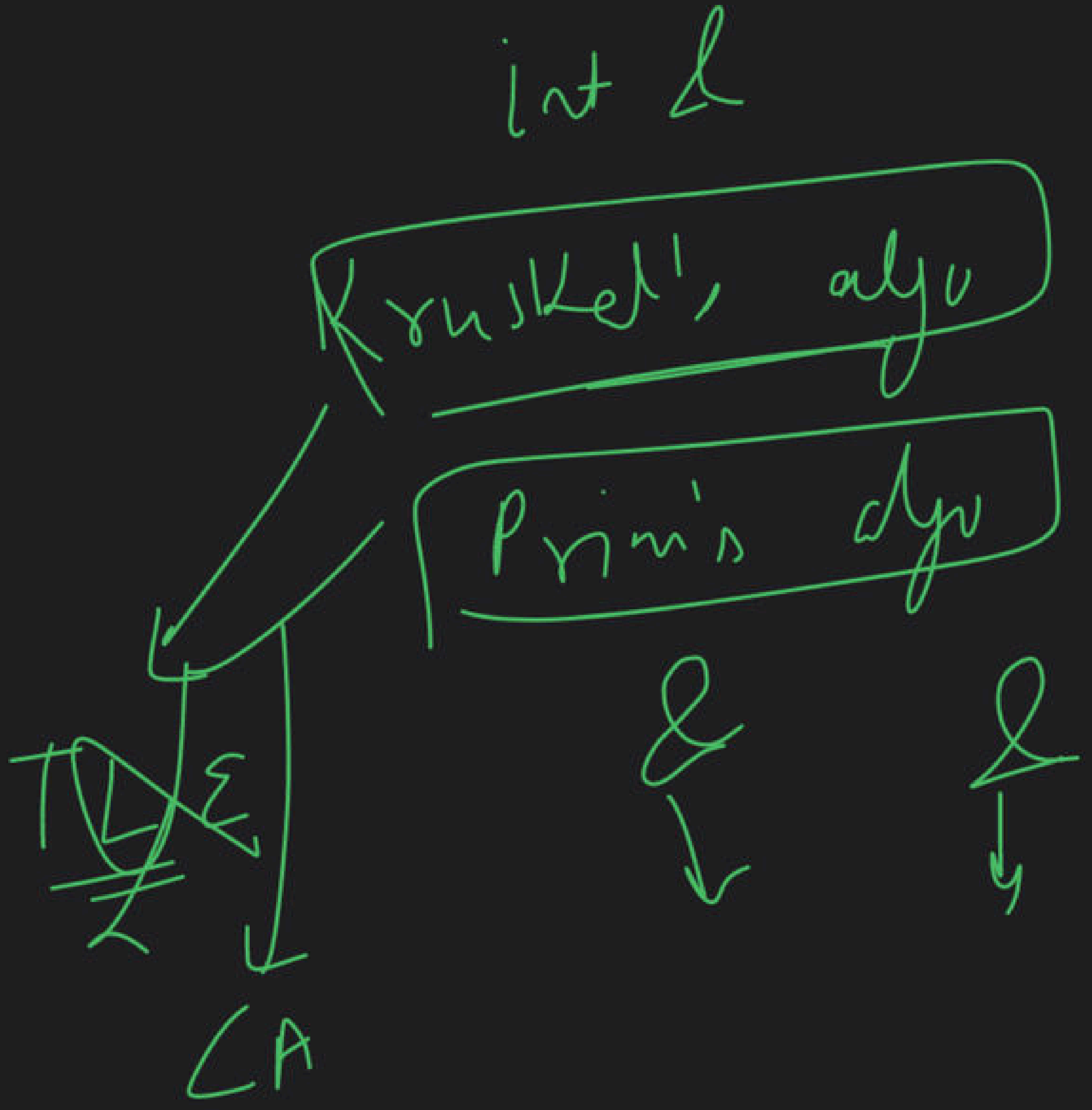
+ /.1

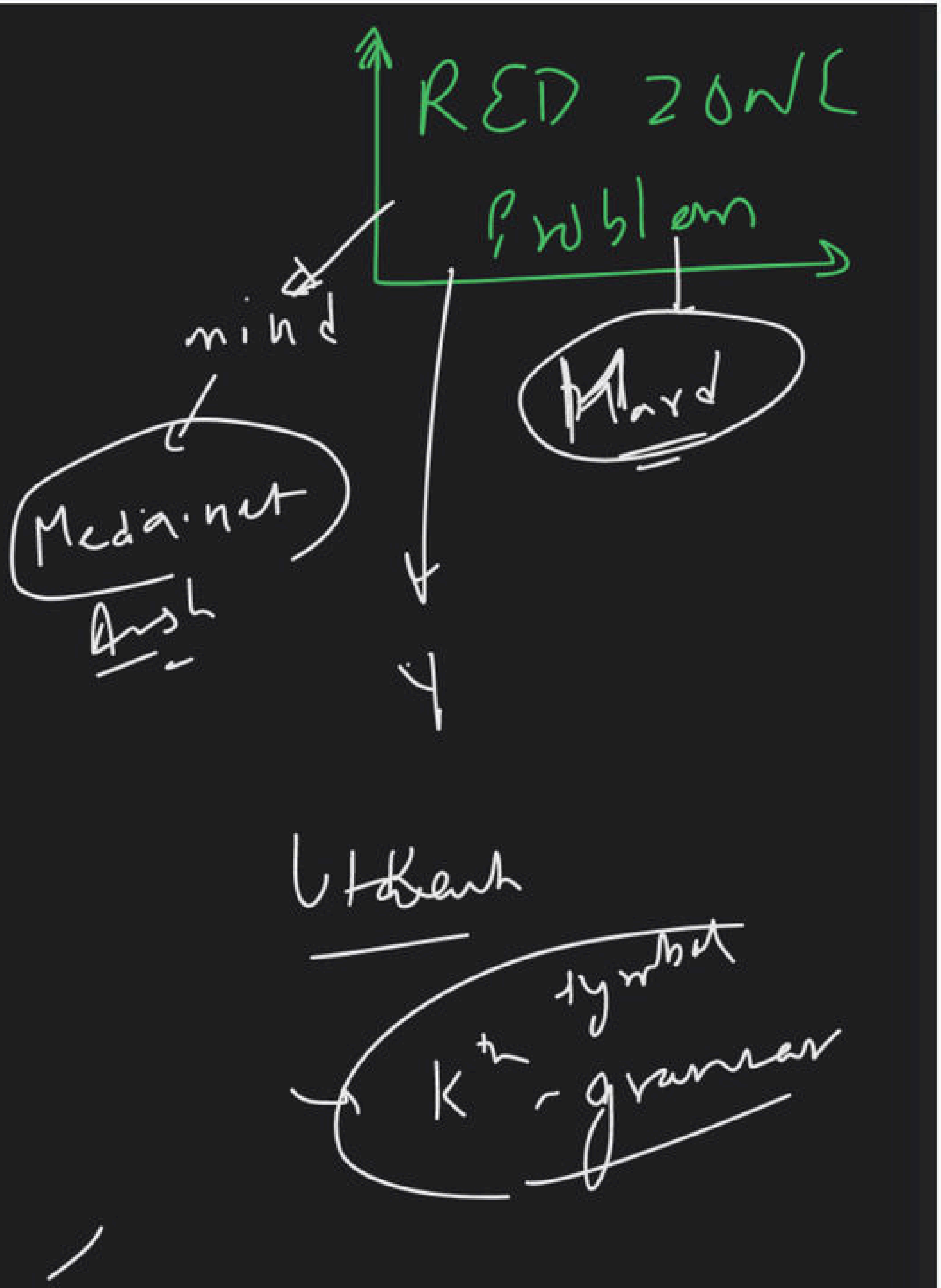
2 -> error → solve

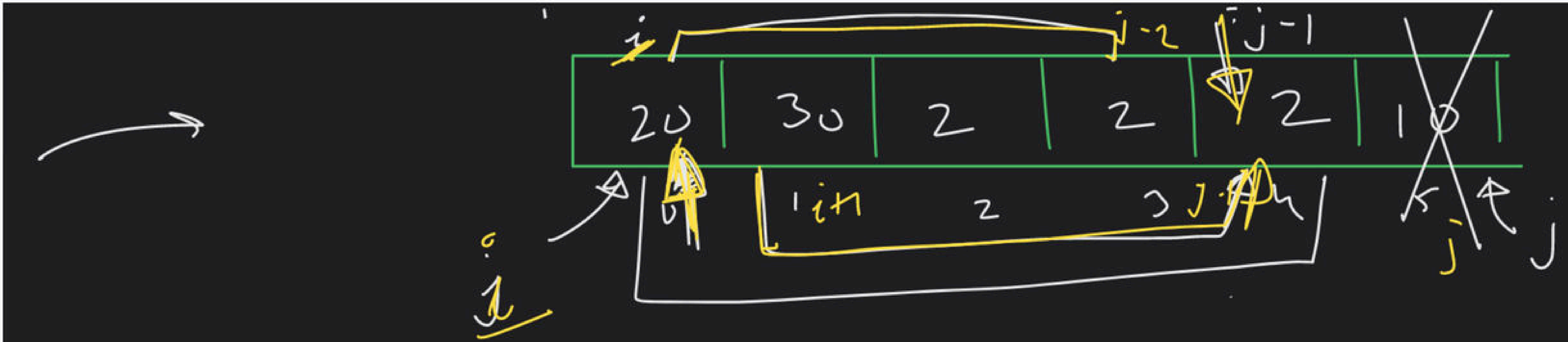
&


3 mins
350
0
1:1
10-20
4-5
1-2
int i = 5
5
i, j
int &j = i
test -1
1-100
100
inclusion
exclusion
Power Set
concept
Leetcode
Com
1 _ 2 _ }
1 _ _ }
1 2 _ }
1 2 }


516, 0
51, 6
5, 61
0, 615
5

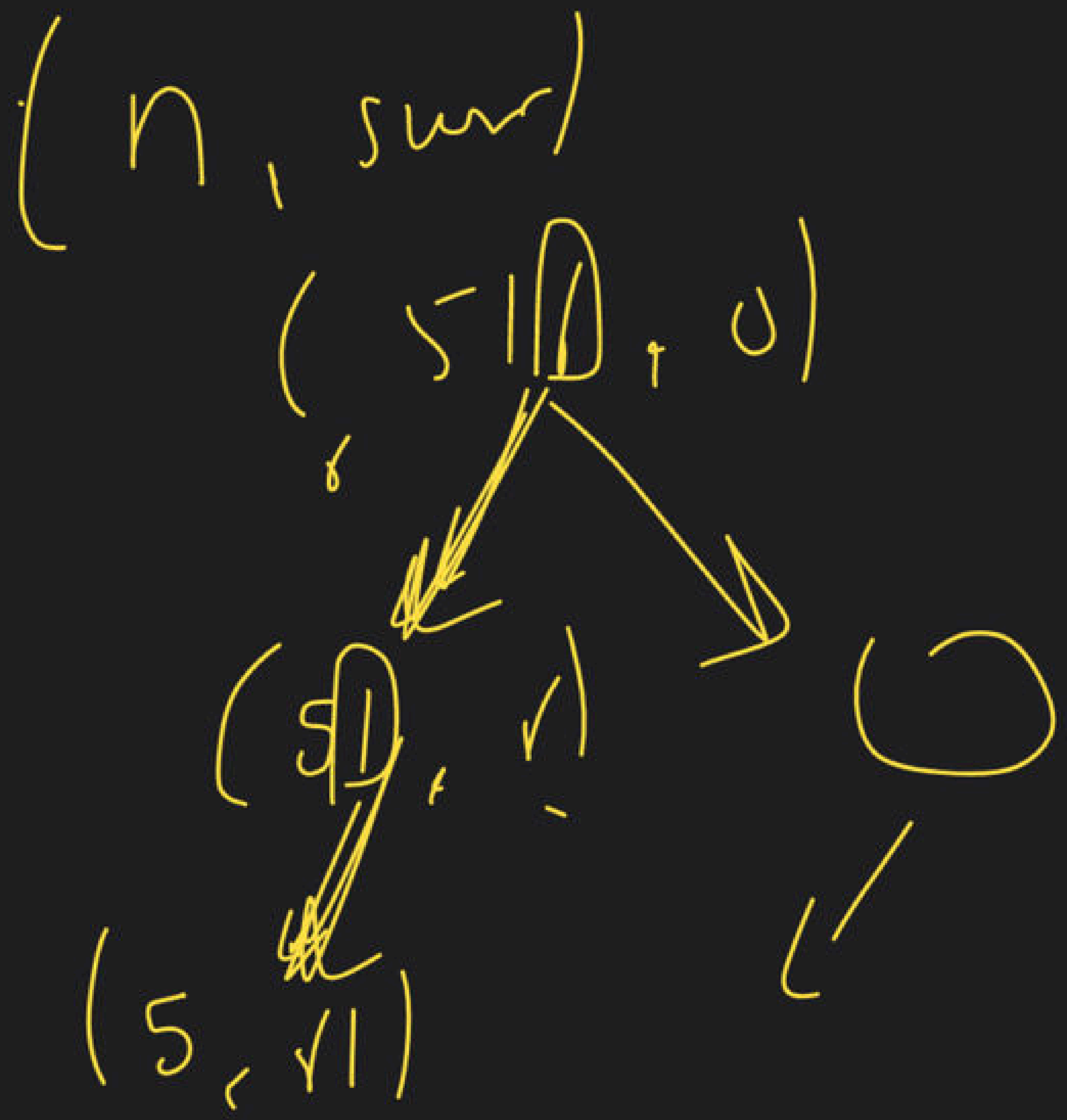

int &
Kruskal's algo
Prim's algo
TLE
CA
&
&


RED ZONE
Problem
mind
Media.net
Ansh
Hard
Y
Utkarsh
Kth symbol grammar



choice1 = $arr[i] + \min\left(f(i+2, j),\ f(i+1, j-1)\right)$

choice2 = $arr[j] + \min\left(f(i+1, j-1),\ f(i, j-2)\right)$



max

Unacademy

Interview Prep BootCamp

DS/DBMS/OOPS/CN

Big

K^{th} character

in grammar

→ n = 1

0

, n = 2

0

0 1

, n = 3

0

0 1

0 1 1 0

n = 4

0

0 1

0 1 1 0

0 1 1 0 1 0 0 1

2min

→ n , K

n → row

K → K^{th} grade

n = 1 , K = 1

n = 2 , K = 2

n = 4 , K = 1

1^{st} row → 0

2^{nd} row → 0 1

3^{rd} row → 01 10

4^{th} → 01 10 10 01

0 → 01

1 → 10

Approach

->

n = 1

1st row → 0

n = 2

1st → 0

2nd → 0 1

n = 3

1st → 0

2nd → 0 1

3rd → 0 1 1 0

Approach :-

if (n = =1 || K ==1) } → B.C

return 0;

int mid = pow (2, n-1) / 2 ; → why?

$2^1 \to$ U d

$2^{1-1} \to 2^0 \to \frac{1}{2} = 0$

$2^{2-1} \to \frac{2^1}{2} = \boxed{1}$

$\frac{2^{3-1}}{2} \to \frac{2^{\cancel{2}}}{2} = \boxed{2}$

Question

Dry Run

Next Class

```
if (n = = 1) || K = = 1)
        return 0;

int mid = pow(2, n-1) /2;

if ( K <= mid)
        return f(n-1, K)

else
        return !f(n-1, K-mid);
```

1:5 hr

1:30min

1:42min

→ Dry Run → Kth char → LB → 0

\ 1h → 1 | June

→ Red zone Problem → V.I DSU 5110U5

→ web dev

→ DSA
revised
sheet

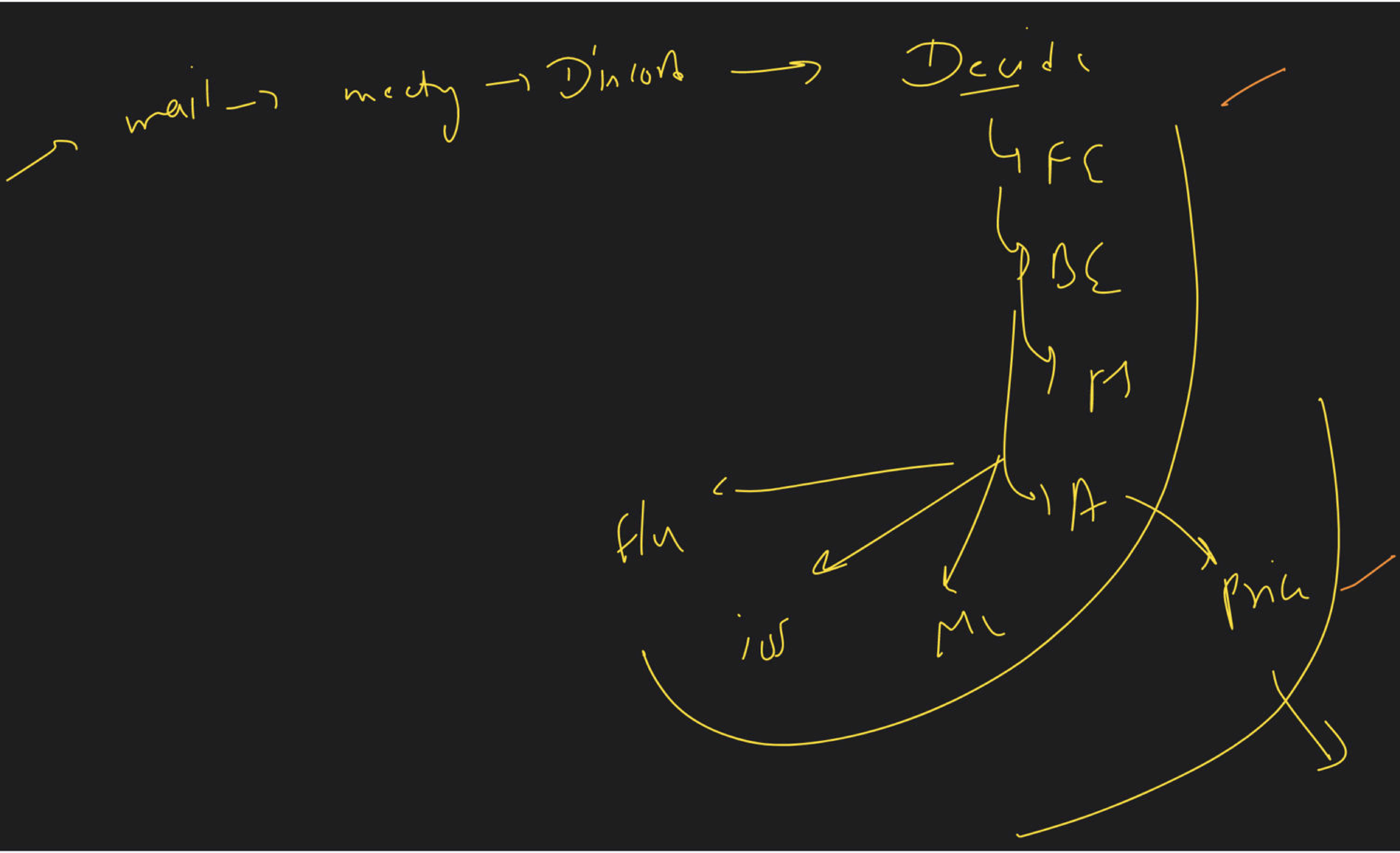
→ MS → ?

→ Recursion → Sheet

→ 318 LC → Max Prod of word lgth

→ Median of 2 sorted arrays → same size

→ diff size


mail
meeting
Discussion
Decide
FE
BE
PM
IA
flu
iOS
ML
price