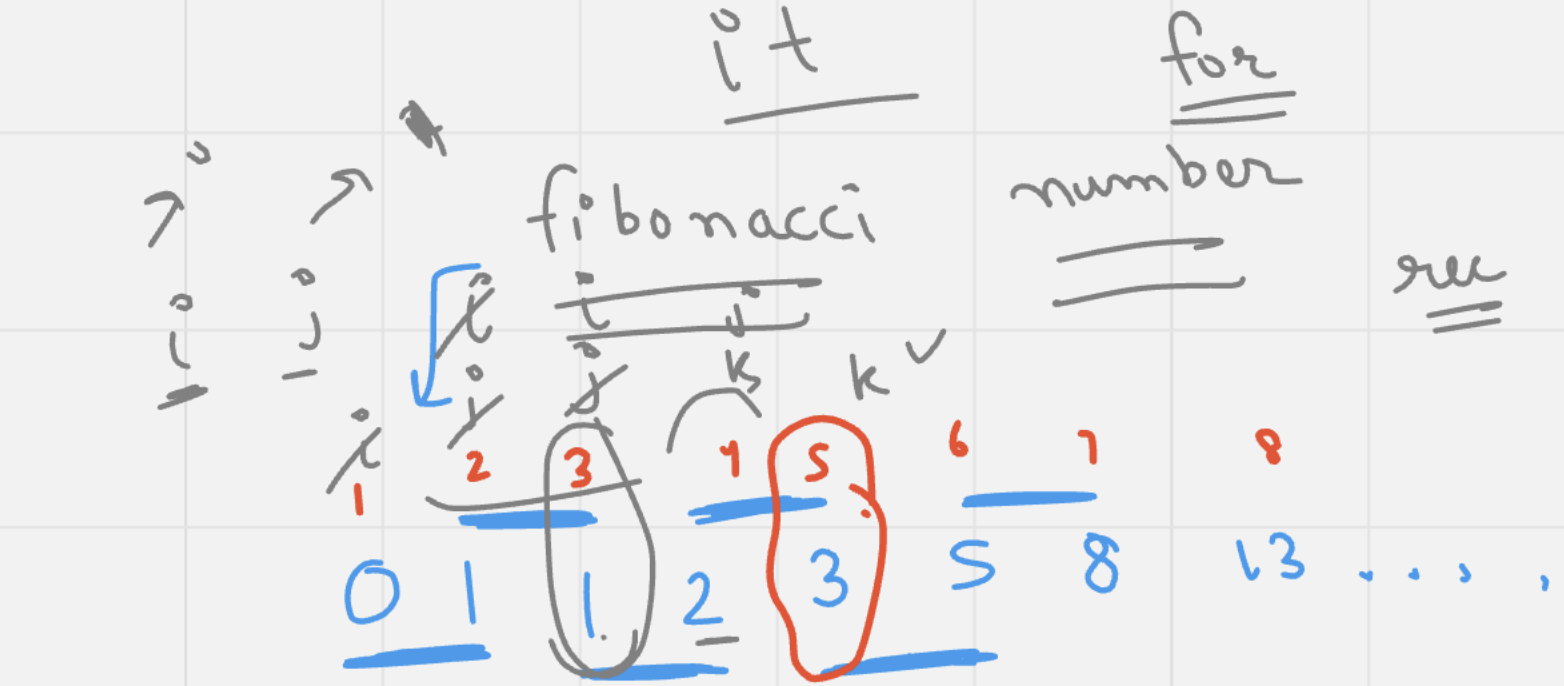


$k = i + j$

0 + 1

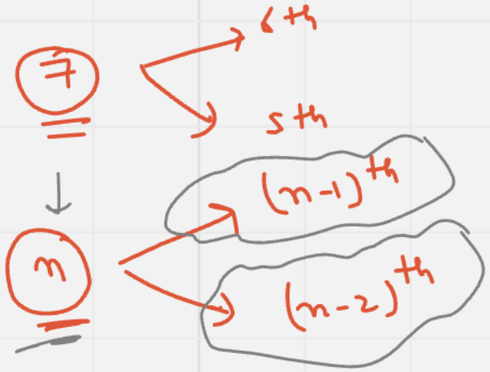
1

nth

5th → 3

8th → 13

1	2	3	4	5	6	7	8
0	1	1	2	3	5	8	13



50th

Fib (n)

[if (n == 1) 0 | if (n == 2) 1] Base

(n-1)th = fib (n-1);

(n-2)th = fib (n-2);

my = (n-1)th + (n-2)th

(2^{50})

$O(2^n)$ → 50 (very big)

2 (very big)

very big

T.C.

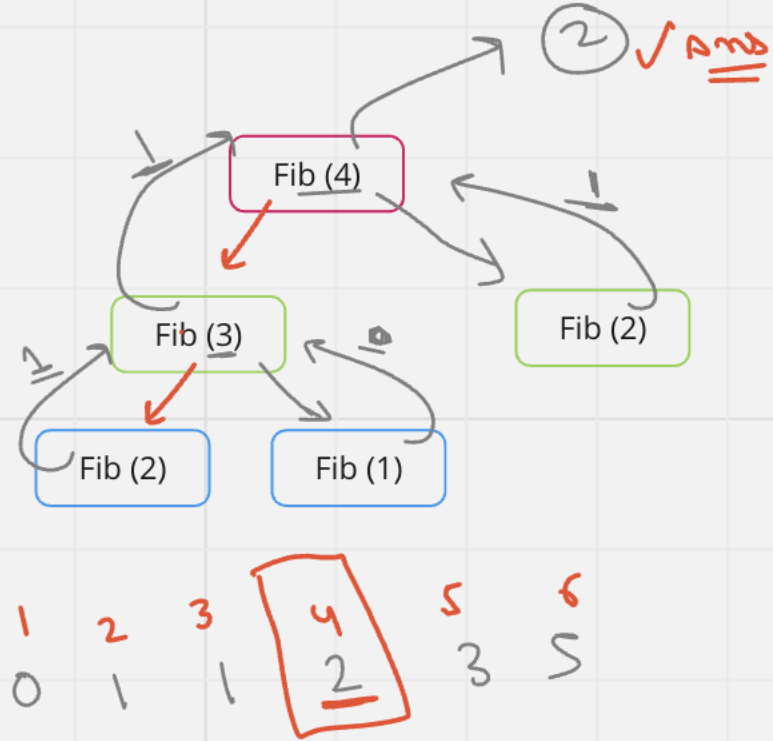
1 sec ≈ 10^8

using Rec

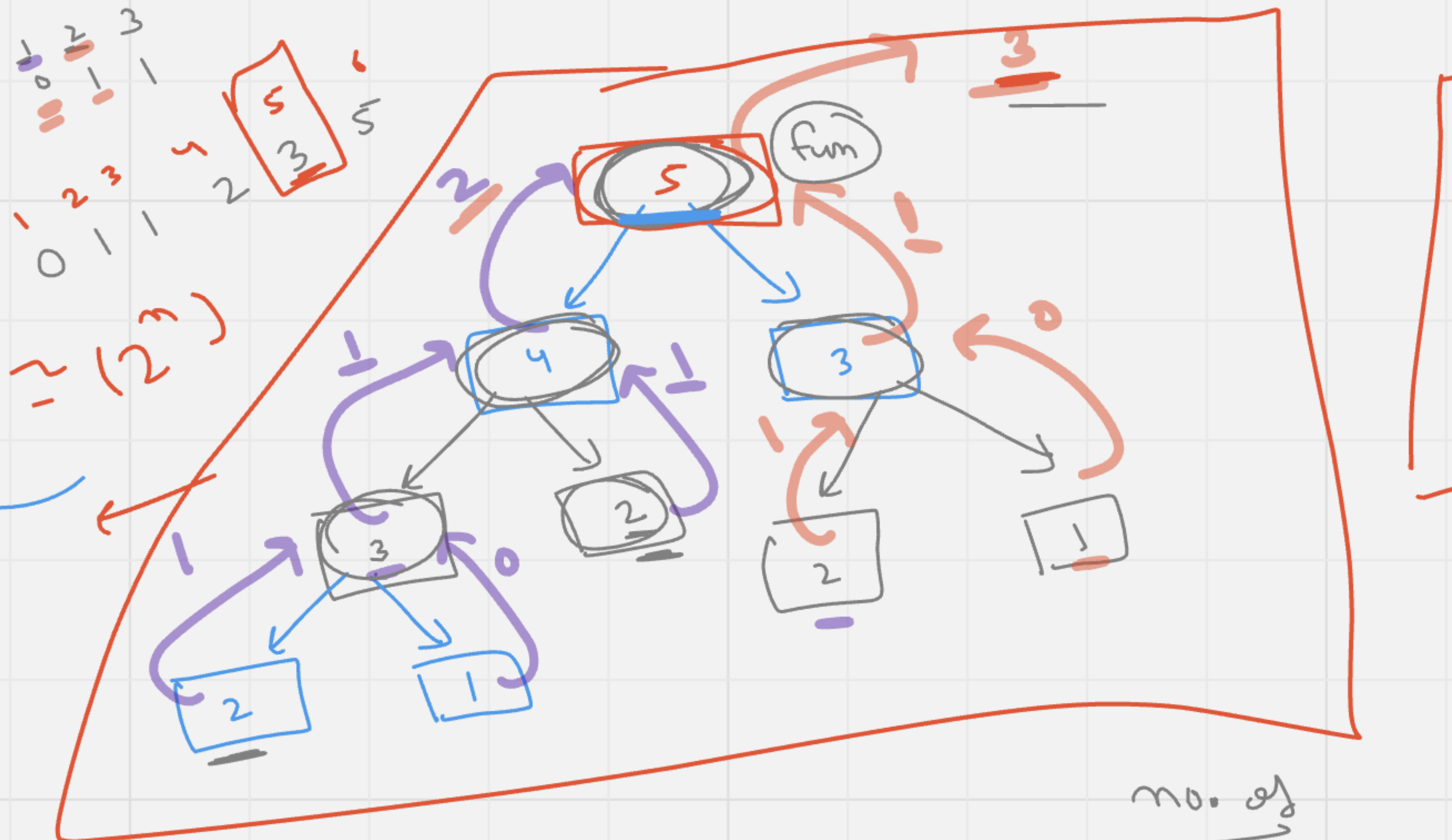
Bigo (2^n)

Big O (2^n)

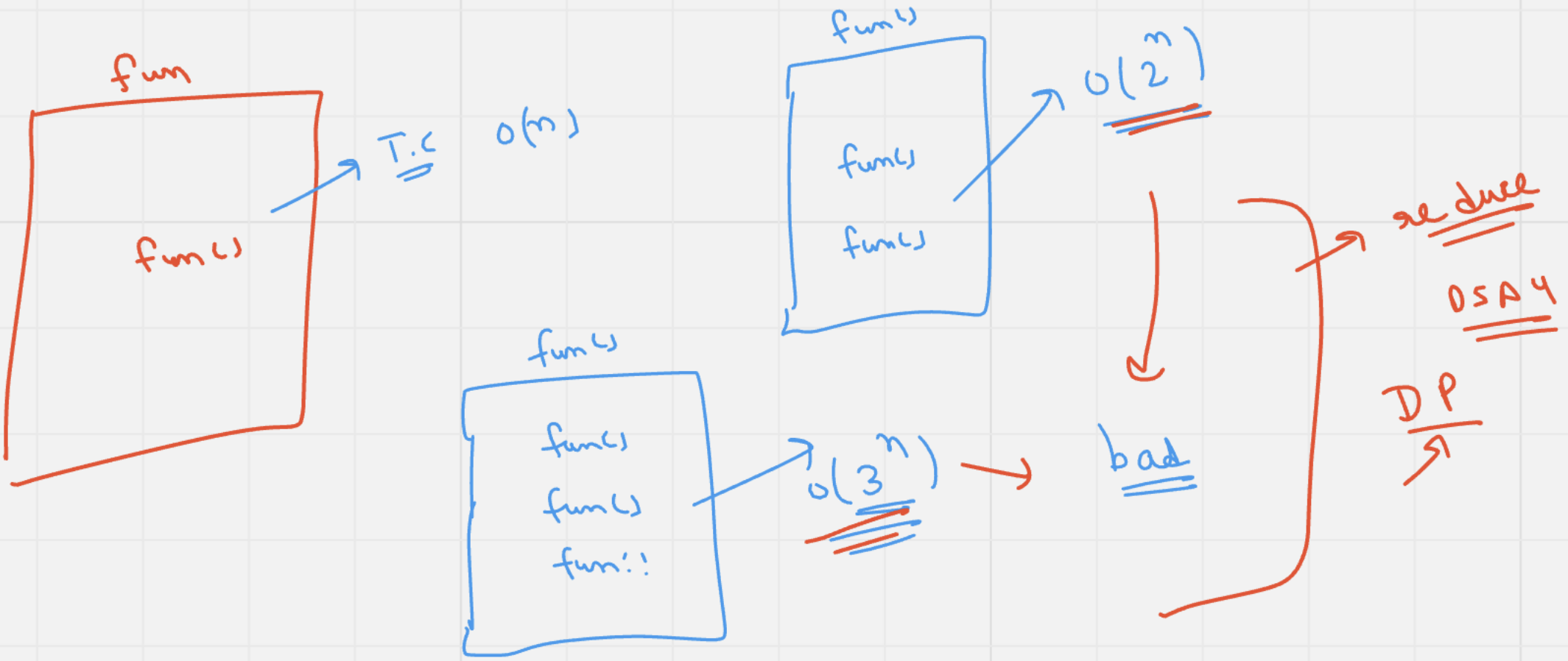
```
public static int fib(int n ){

    if(n == 1) {
        return 0;
    }

    if(n == 2) {
        return 1;
    }

    int fib_n_1 = fib(n-1);
    int fib_n_2 = fib(n-2);

    int myAns = fib_n_1 + fib_n_2;

    return myAns;
}
```



deriv $\approx (2^n)$



T.C. = Sum (no. of fun. call) ← unit Operation


fun
fun()
T.C
O(n)
fun()
fun()
fun()
O(2^n)
fun()
fun()
fun()
O(3^n)
bad
reduce
DSAY
DP

n level

n

n-1

n-2

n-2

1

base case

2^0 1

2^1 2

2^2 4

2^3 8

1 + 2 + 4

n terms

1 + 2 + 4 + ... + n terms

$1 + 2^1 + 2^2 + \dots + n\text{terms}$

G.P.

a = 1 r → 2 n

Sum of first n terms of GP Formula

$$S_n = \frac{a(1 - r^n)}{1 - r} \quad ; r \neq 1 \text{ and } r < 1$$

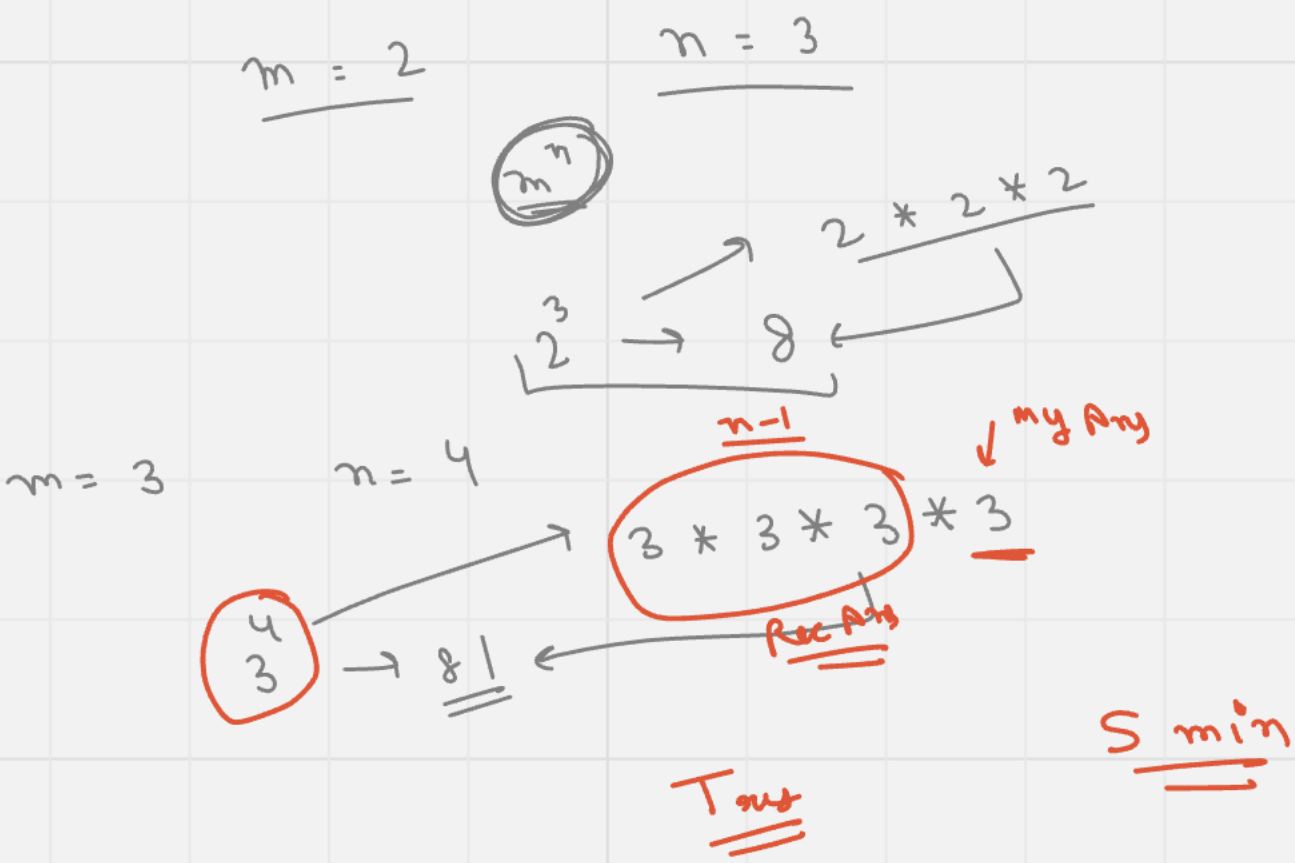
$$S_n = \frac{a(r^n - 1)}{r - 1} \quad ; r \neq 1 \text{ and } r > 1$$

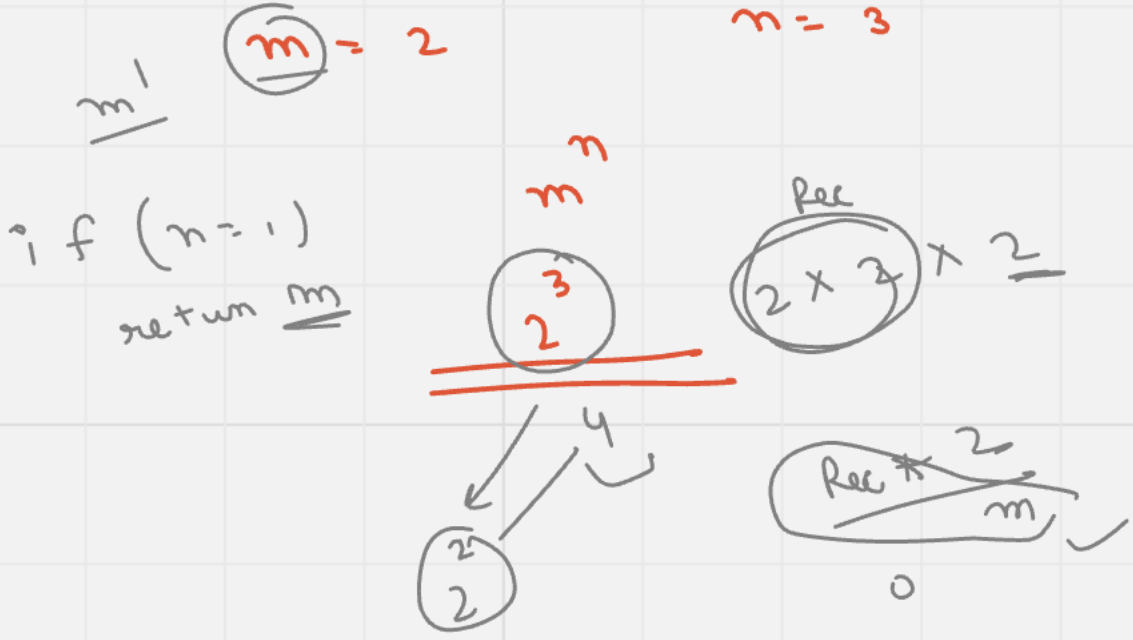
$$S_n = \frac{1(2^n - 1)}{2 - 1}$$

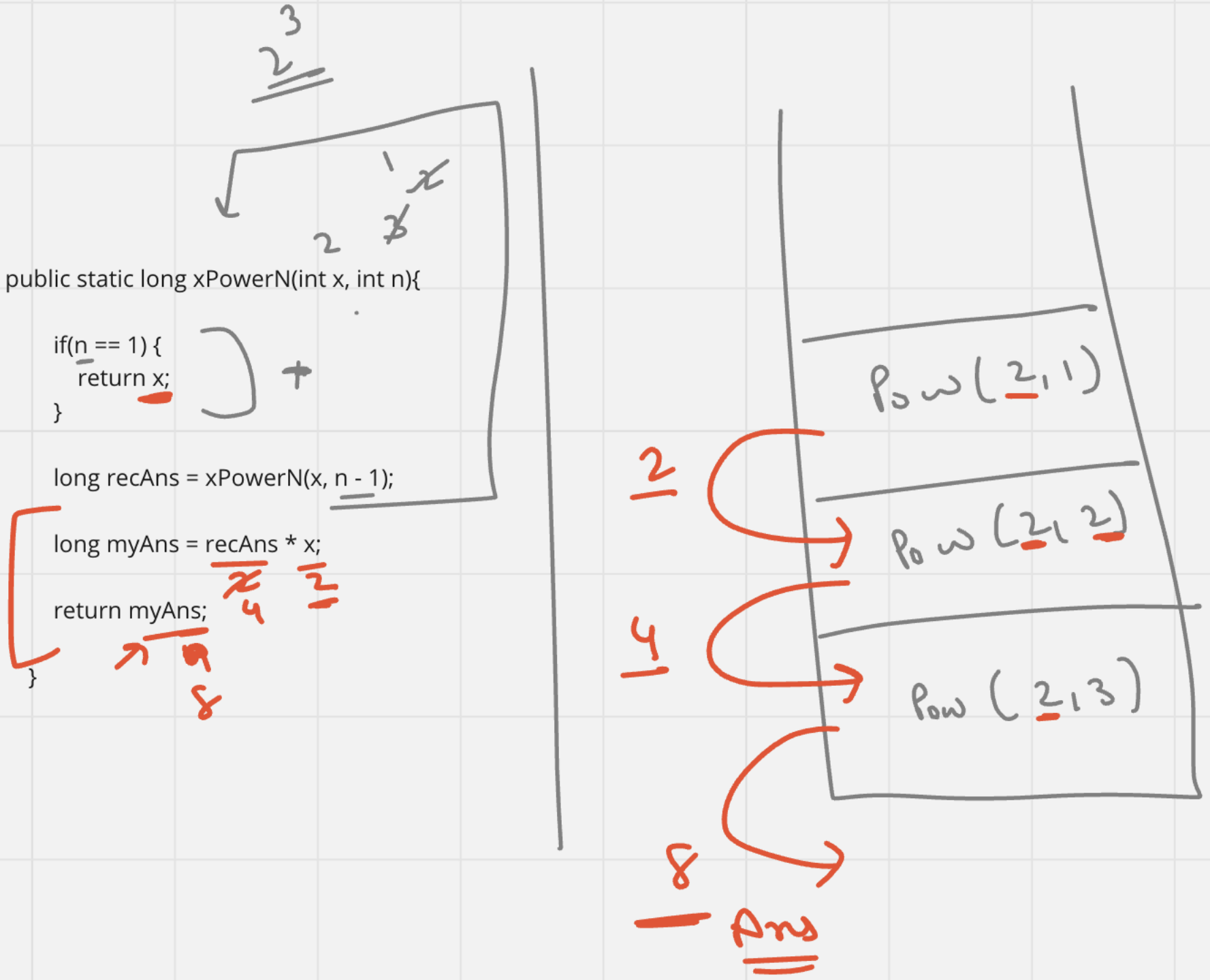
$$= 2^n - 1$$

2^n

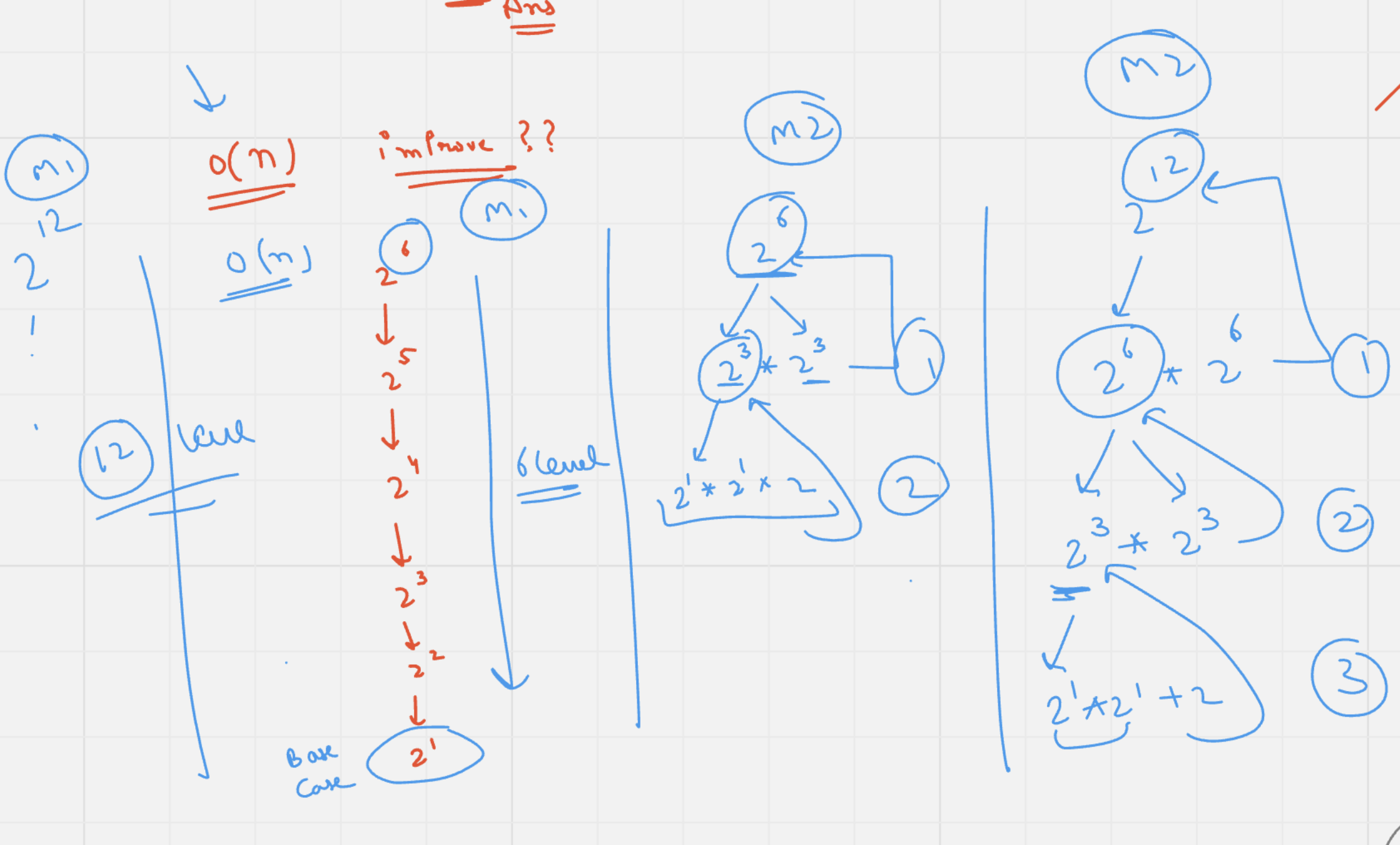
T.C

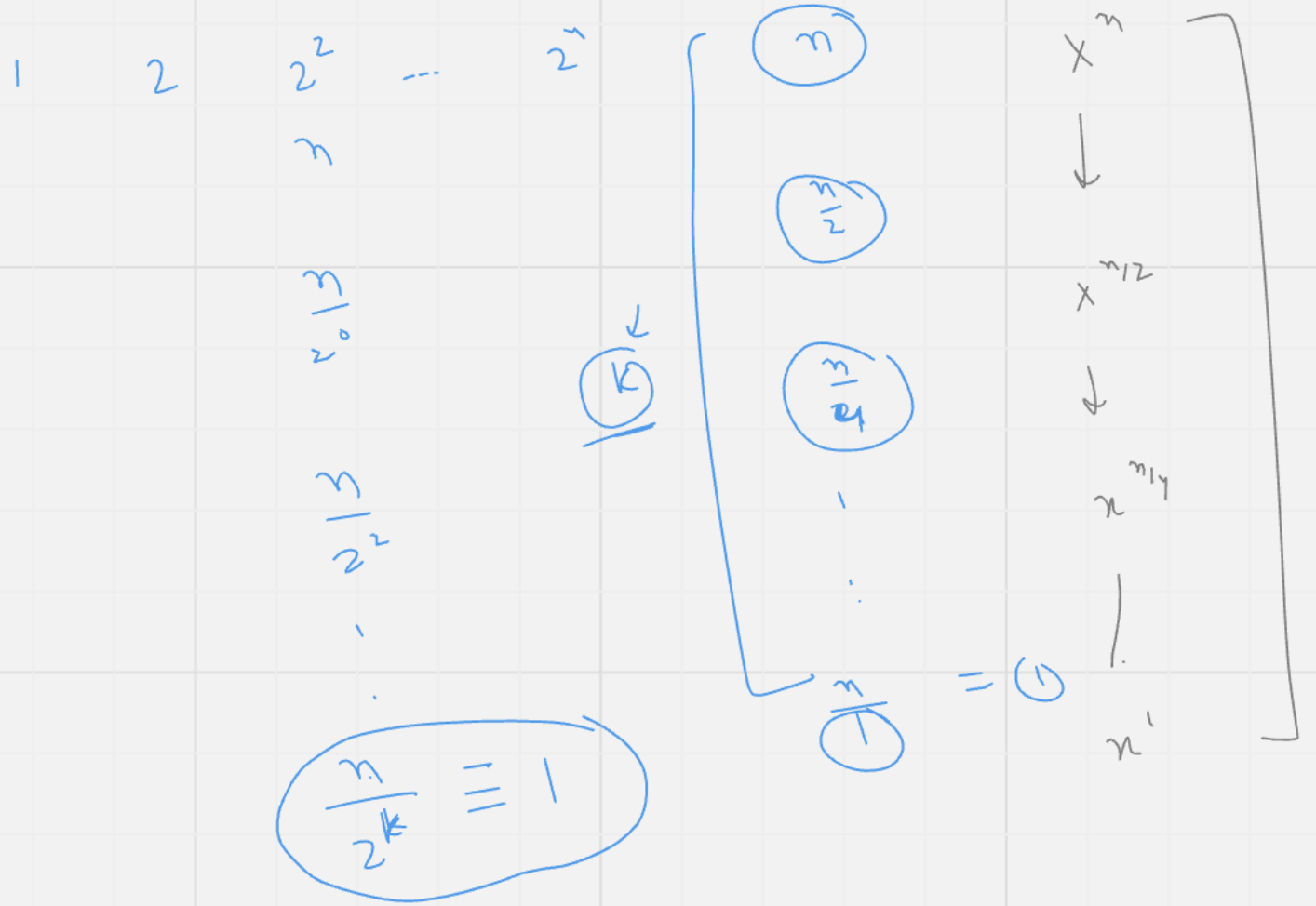

m = 2
n = 3
m n
2 * 2 * 2
2 3
8
m = 3
n = 4
n-1
my Any
3 * 3 * 3 * 3
Rec Any
4 3
81
S min
T out


m = 2
n = 3
m!
if (n == 1)
return m
m n
Rec
2 X 3 X 2
3 2
4
2 2
Rec * 2
m
0



```
public static long xPowerN(int x, int n){

    if(n == 1) {
        return x;
    }

    long recAns = xPowerN(x, n - 1);

    long myAns = recAns * x;

    return myAns;
}
```


M1
2^{12}
O(n)
O(n)
12 level
improve ??
M1
6
2^6
2^5
2^4
2^3
2^2
2^1
Base Case
6 level
M2
6
2^6
$2^3 * 2^3$
1
$2^1 * 2^1 * 2$
2
M2
12
2^{12}
$2^6 * 2^6$
1
$2^3 * 2^3$
2
$2^1 * 2^1 + 2$
3



$n = 2^k$

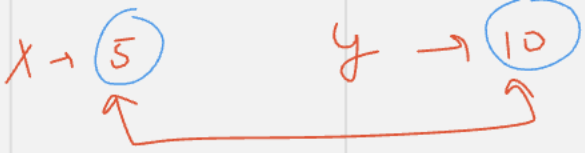
Take Log

$\log n = \log(2^k)$

$\log n = k \log_2 2$

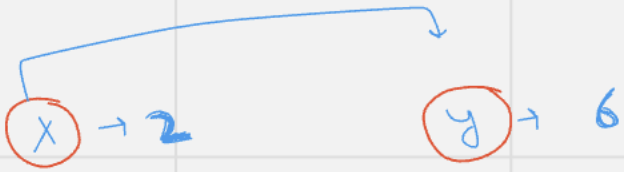
$\log_2 2 = 1$

$k = \log n$



5 6 7 8 9 10

output



2 3 4 5 6

Recursion

x 2 → y → 6 iterative

[ ]

x == y

print

rec

my Ans

x

y - 1

2 3 4 5 6

y

Correct

12th fib

5th

15

Even

3 fib 6th 9th 12th

1 2 3 4 5 6 7 8 9 10 11 12

1 1 2 3 5 8 13 21 34 55 89 144

2nd Even fib = 8 ≃ 2*3 = 6th fib

4th Even fib = 144 ≃ 4*3 = 12th fib

1st Even fib = 2 ≃ 1*3 = 3rd

nth fib → (n-1) + (n-2)

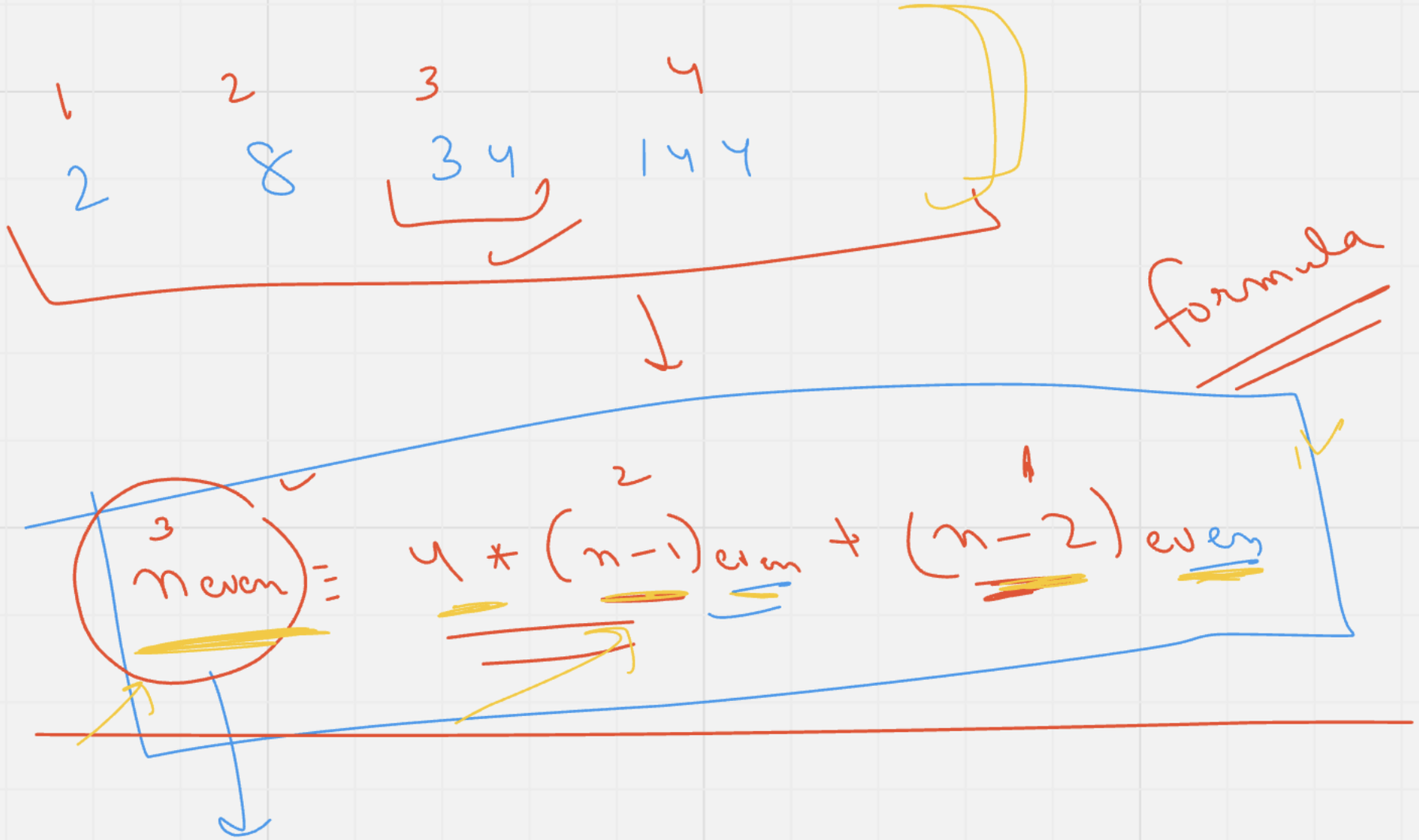
50th TLE

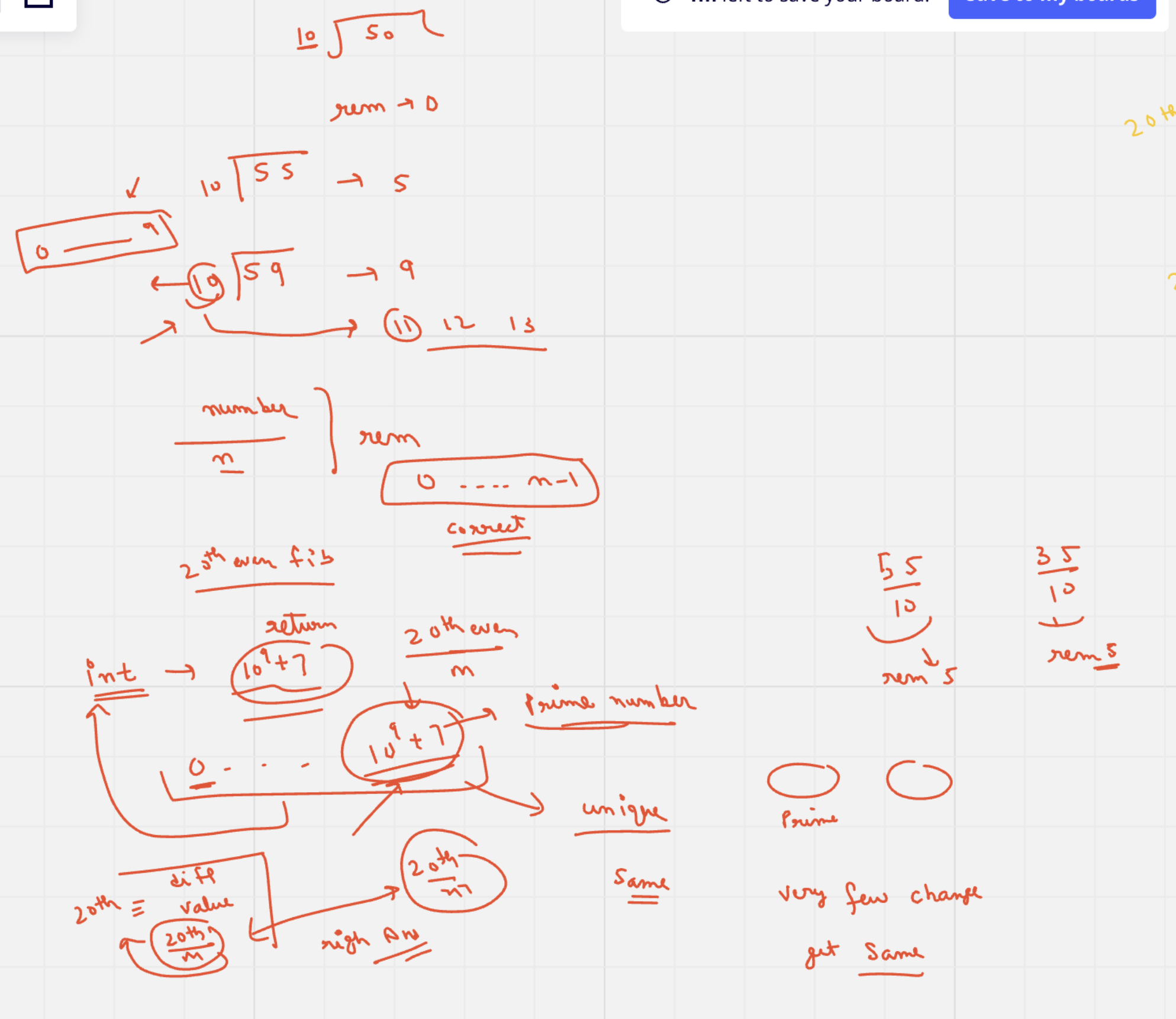
5 Even fib ≡ 5*3 ≡ 15th fib

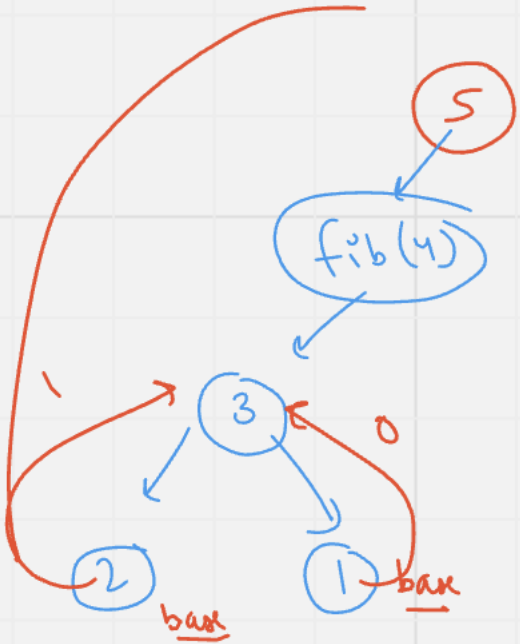
20th Even fib ≡ 20*3 = 60th fib

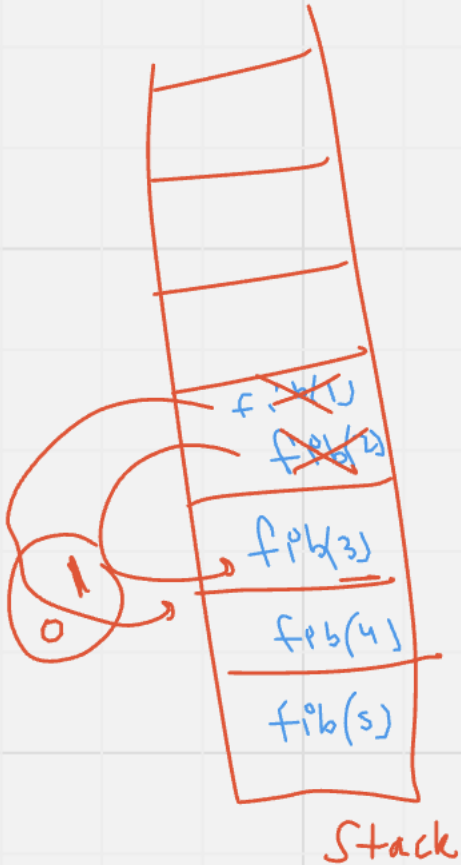
TLE no

O(2^n) vv bad


1 2 3 4
2 8 3 4 1 4 4
formula
how
3 even = 4 * (n-1) even + (n-2) even
3 even = 4 * 8 + 2
= 32 + 2 = 34


10 | 50
rem → 0
10 | 55 → 5
0 —— 9
19 | 59 → 9
11 12 13
number
n
rem
0 .... n-1
correct
20th even fib
int →
return
10^9+7
20th even
n
prime number
10^9+7
0 - - -
unique
same
20th
n
diff
20th ≡ value
20th
n
right Ans
55
10
rem 5
35
10
rem 5
Prime
very few change
get Same


5
fib(4)
1
3
0
2
1
base
base


fib(3)
1
0
fib(4)
fib(5)
Stack