Binary Search (Smart)

Kahan Lagega → sorted search space

Kyun hota → T.C. sudhaar deta hai

$n \rightarrow \log n$

$n^2 \rightarrow n \log n$

arr = { 2, 71, 3, 814, -6, -3 } tar = -6

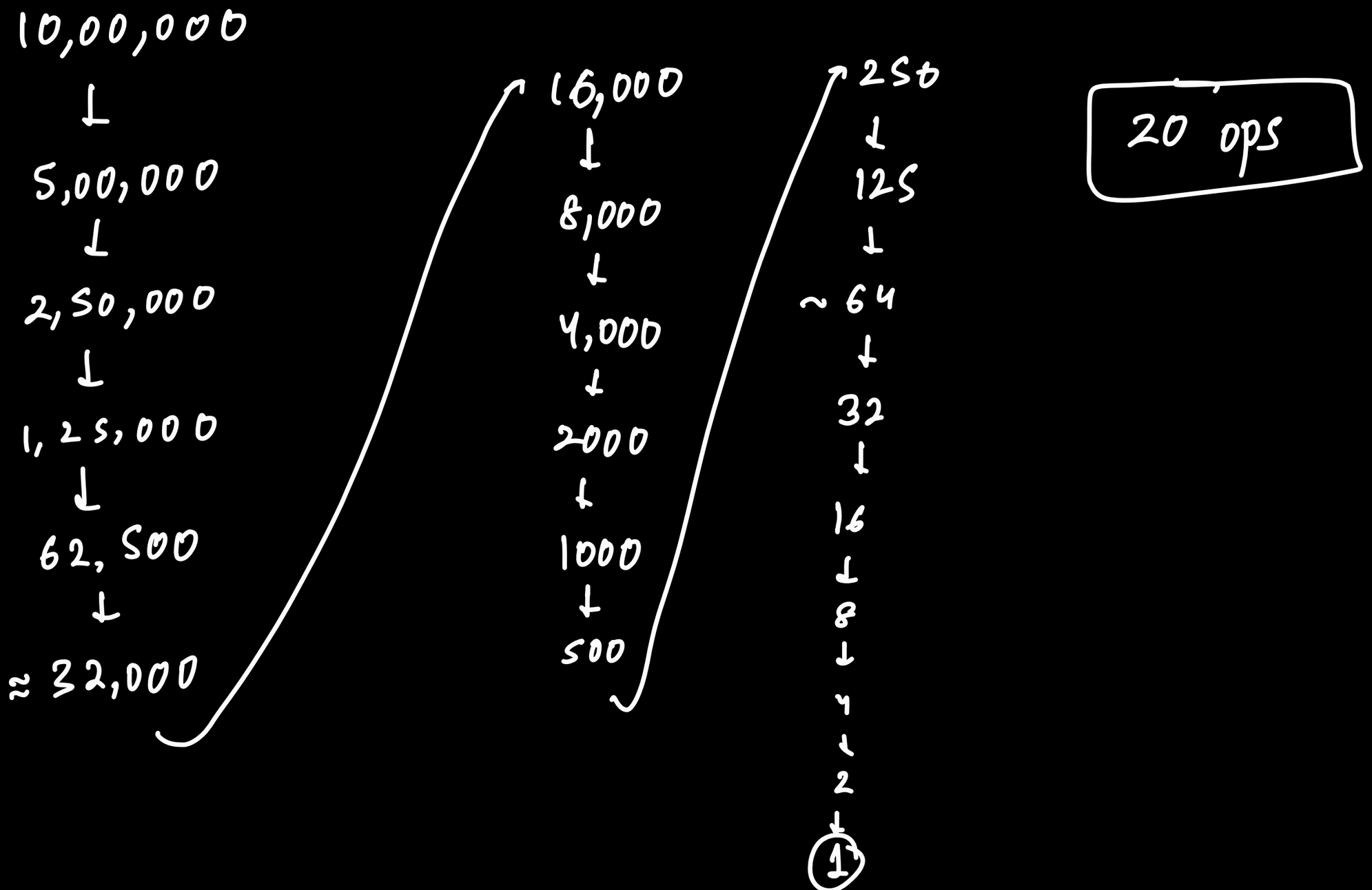
arr = { -76, -4, 9, 28, 47, 49, 510, 615, 9911, 99999 }

tar = 510

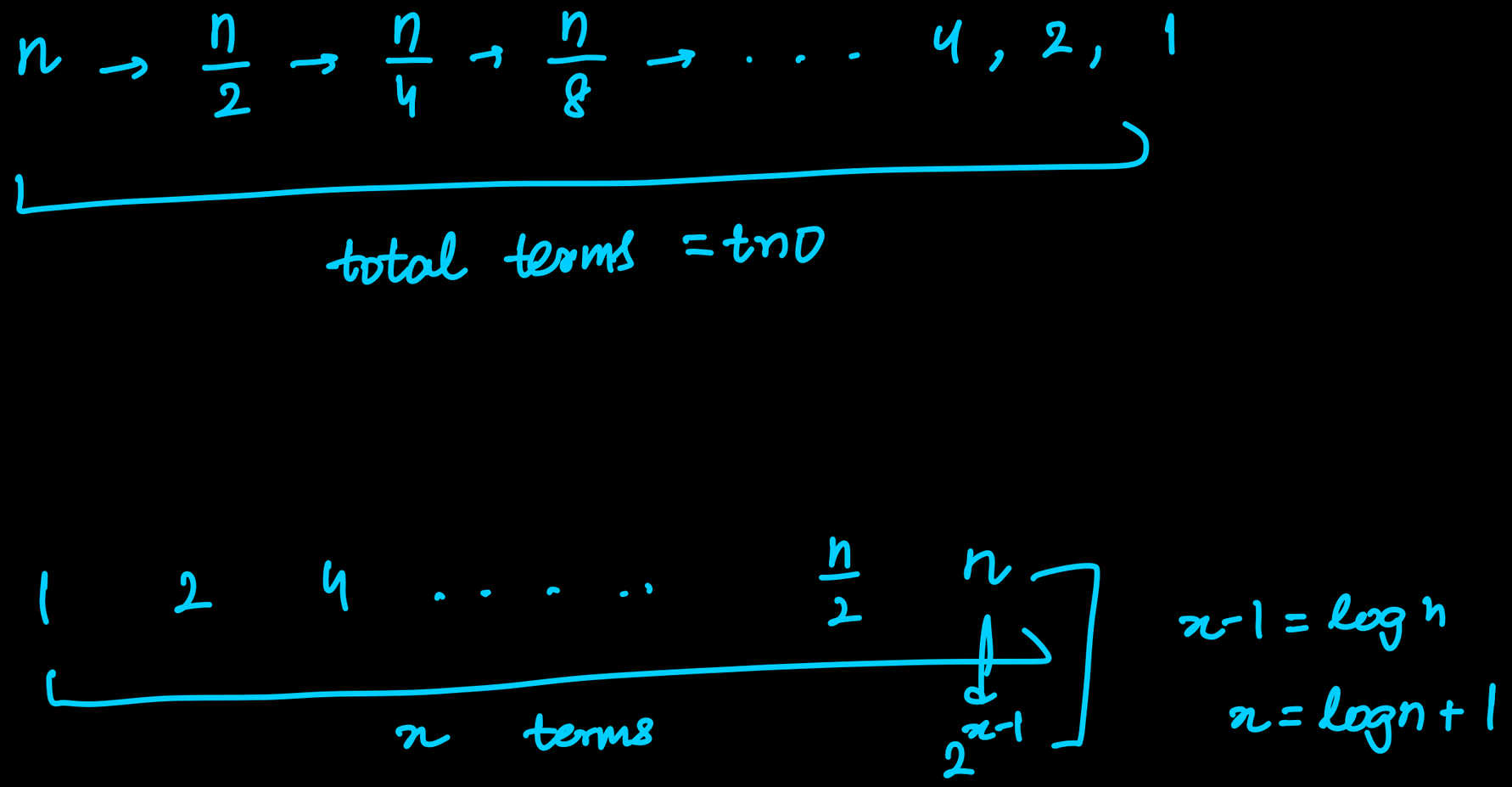
if(mid > tar) go left

if(mid < tar) go right

if (mid == tar)


10,00,000
5,00,000
2,50,000
1,25,000
62,500
≈ 32,000
16,000
8,000
4,000
2000
1000
500
250
125
~ 64
32
16
8
4
2
1
20 ops

Time Complexity



$$T.C. = O(x) = O(\log n)$$

Code

tar = 9

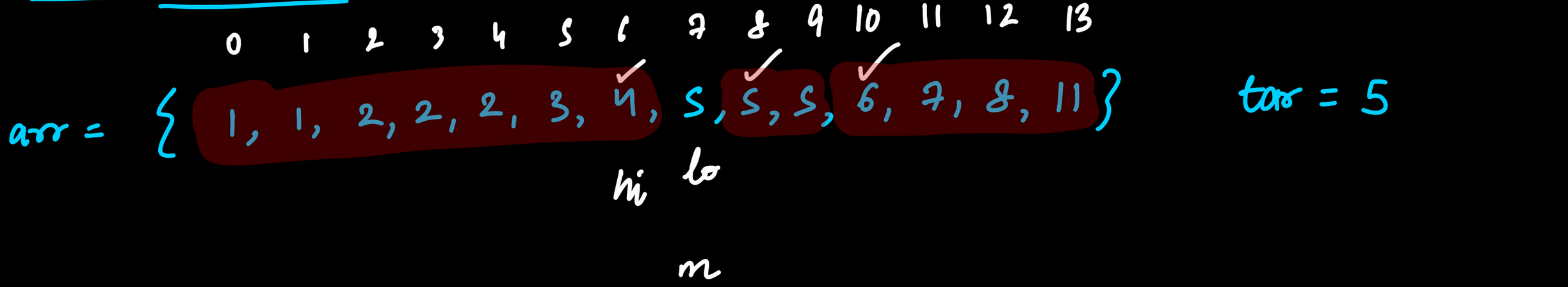
arr = { -76, -4, 9, 28, 47, 49, 510, 615, 9911, 99999 }

lo hi

m

```
int lo = 0, hi = n-1;
while (lo <= hi){
    mid = (lo+hi)/2
    if (arr[mid] > tar)  hi = mid-1
    else if (arr[mid] < tar ) lo = mid + 1
    else (mil gaya)
}
```

First Occurence



index = ~~1~~ 8 7

Algo → if(mid < tar) go right
if(mid > tar) go left
if(mid == tar) mark & go left

Homework : First & Last Occurrence

Homework : Search in descending order array

Ques: Find Peak in a mountain array.

{ -1, 0, 1, 2, 5, 6, 8, 6, 3 }

peak → arr[i] > arr[i+1] && arr[i] > arr[i-1]