Q. Given q queries.

TRIES

Type 1 → store x

Type 2 → return the K^{th} smallest no.

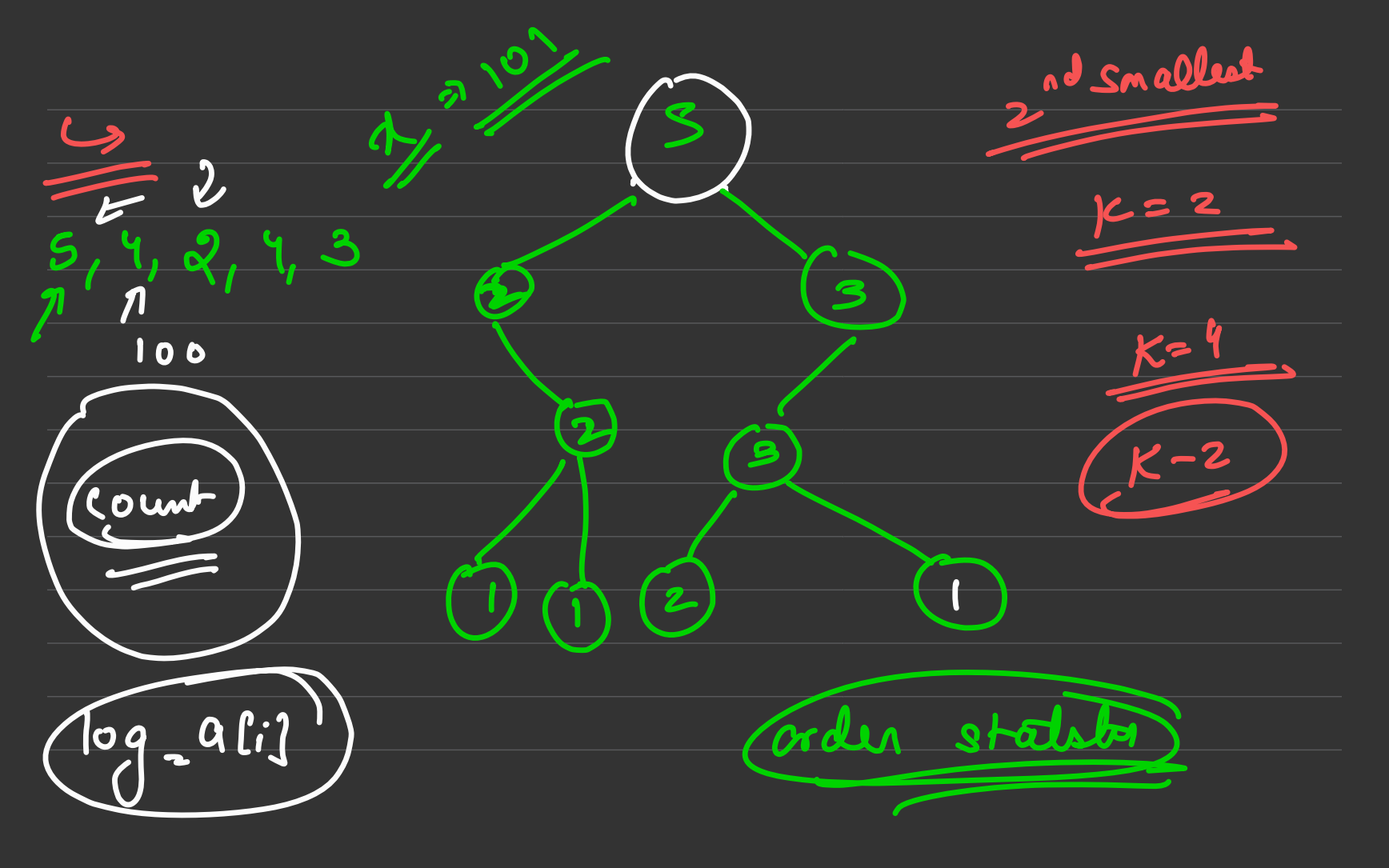
↳ store 1

store 2

store 5

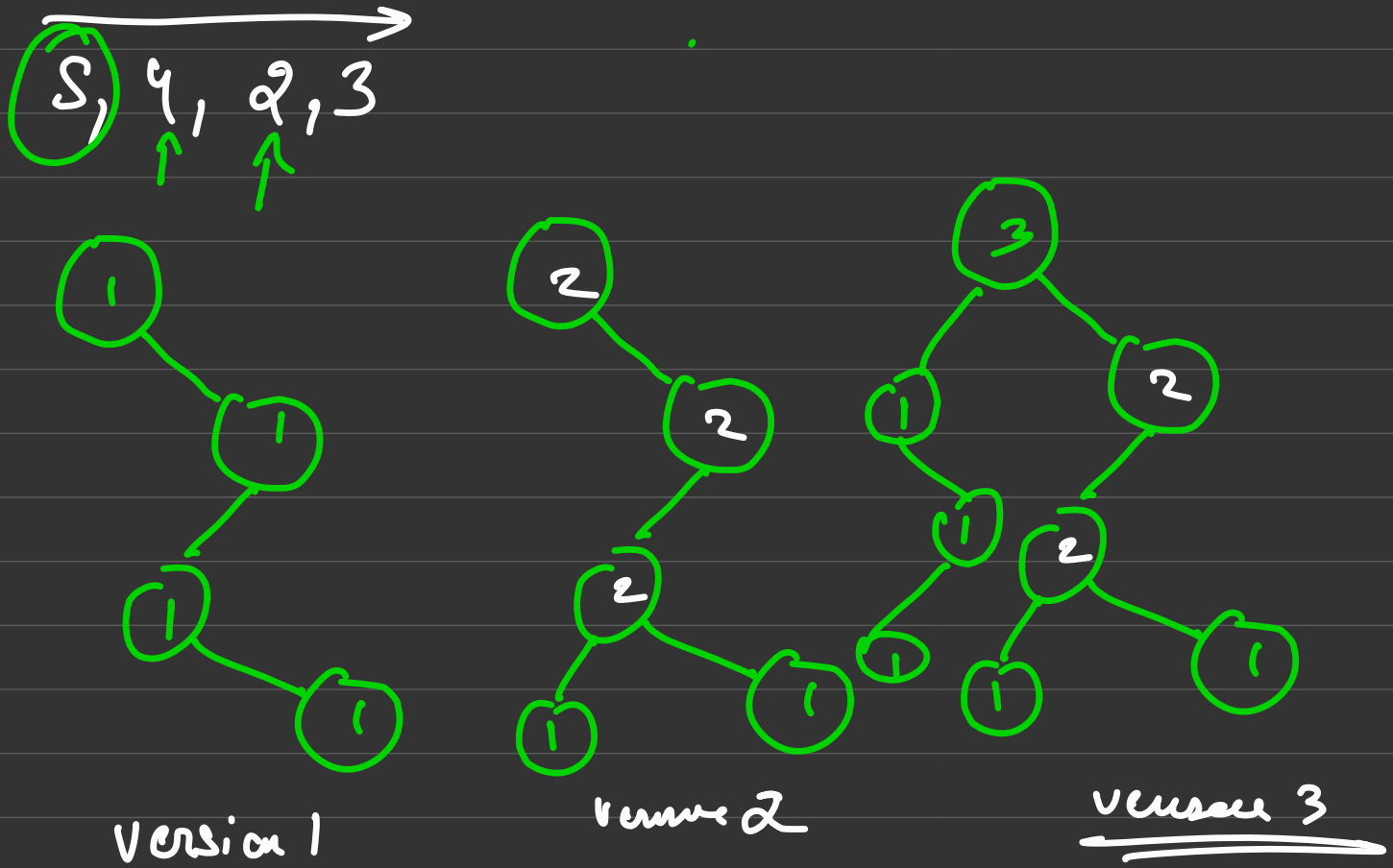
store 9

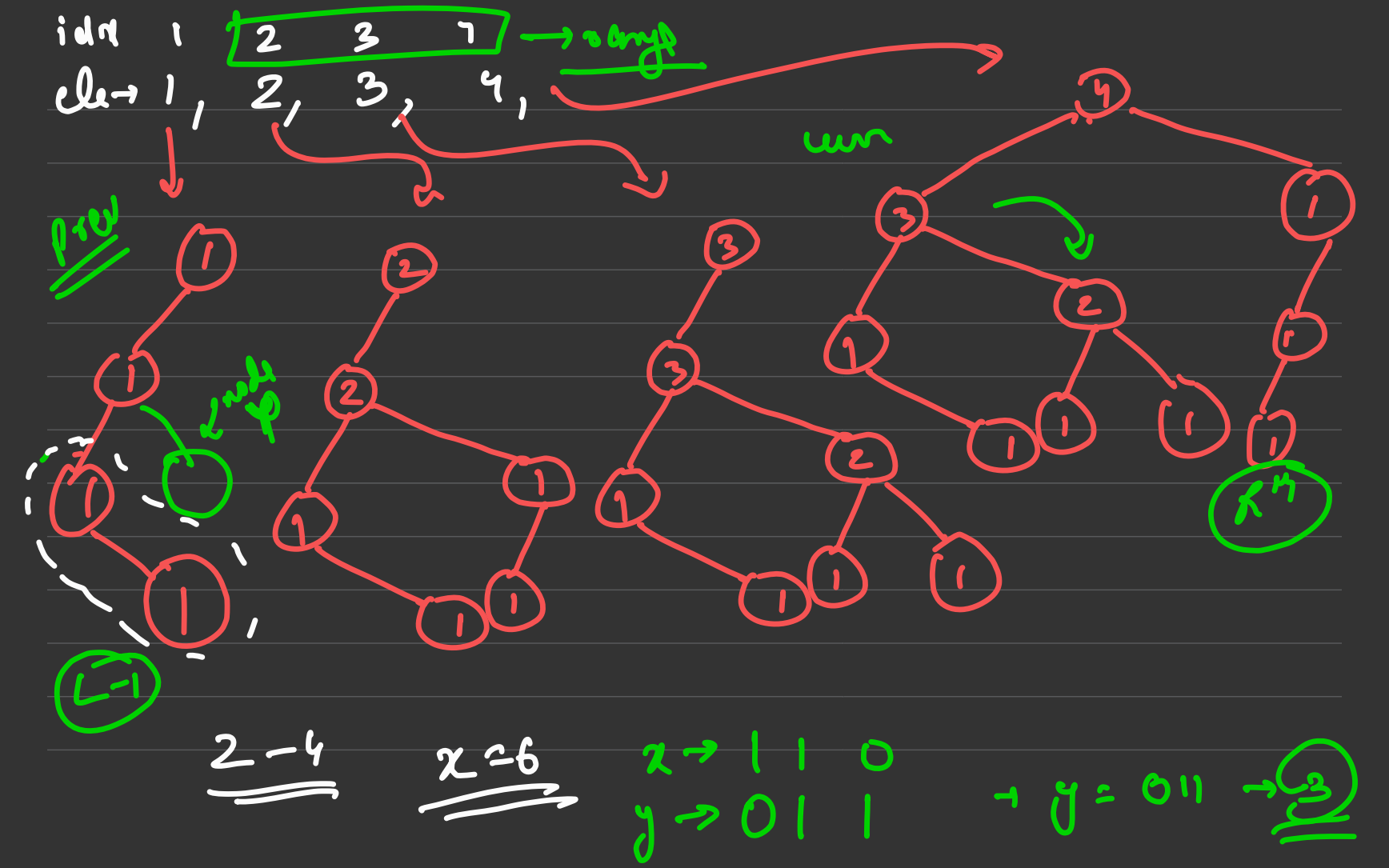
3^{rd} smallest → 5


2
5, 4, 2, 4, 3
100
count
log_a[i]
3
2
3
2
3
1
1
2
1
2nd smallest
K = 2
K = 4
K-2
order statistics

Persistent DS → It helps us to preserve an older state of our data structure

Brute force
Solution would be
persistent i.e


S, 4, 9, 3
1
1
1
1
2
2
2
1
3
1
2
1
2
1
1
1
1
1
Version 1
Version 2
Version 3



2-4

x=6

x → 1 1 0

y → 0 1 1

→ y = 011 → 3

$$p(d,x) = p(0,x) - p(0,d-1)$$

→ Problems →

a) Whole trie is copied again & again

2^n nodes → copy

b) There is some part which repeats unnecessarily

c) we have some part that has relevance after copy.

→ How about instead of copying a whole tree, can we copy only reqd nodes.

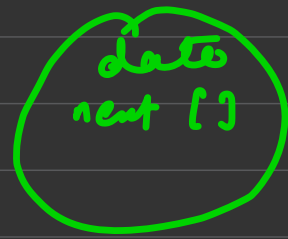
Persistent Trie

$x = 6 \rightarrow 110$

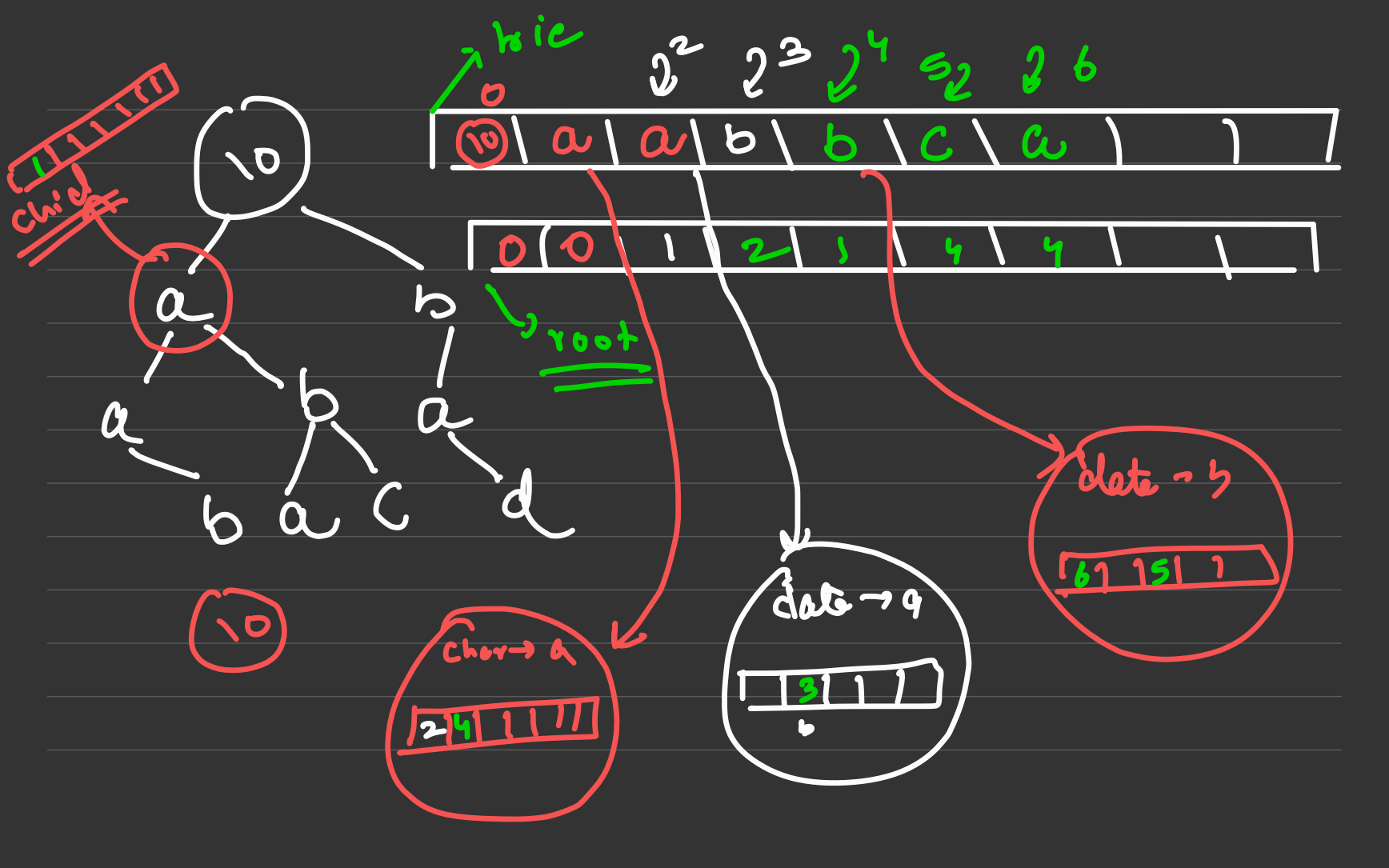
$011 \rightarrow 3$

2 Implementation of trie

1. (Node based) →



2. Array Based


root
10
a
b
a
b
a
c
a
b
a
c
d
10
Char → a
2 4
data → 9
3
data → 5
6 15 1