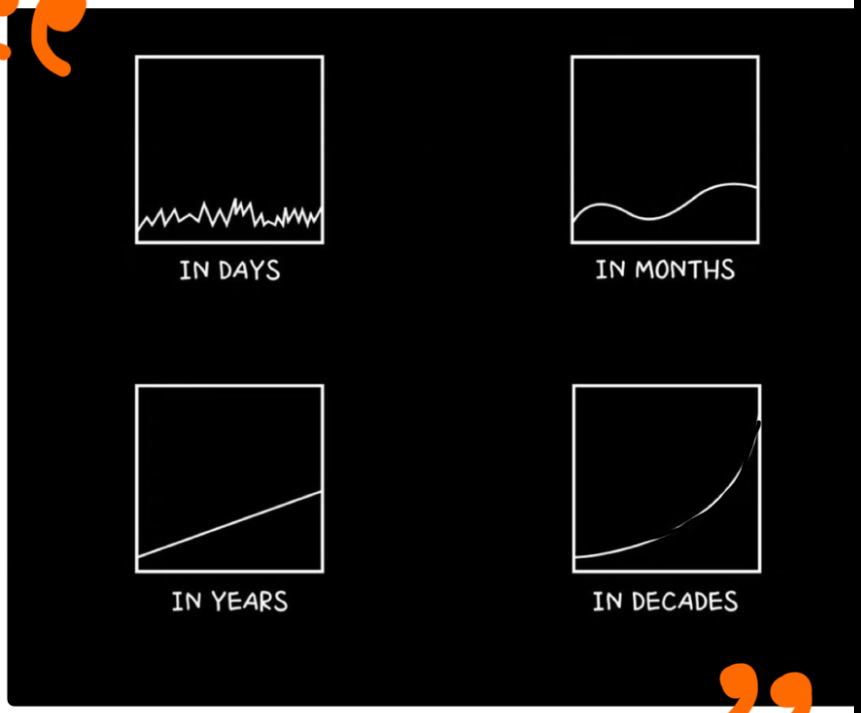
Recursion Concepts & Qns ...

Motivation (भाषण) ...



video 16

#codestorywithMIK

Facebook
Instagram → codestorywithMIK

(Twitter) → CSwithMIK

codestorywith MIK →

Company :- snapdeal

Power Set

Medium Accuracy: 43.3% Submissions: 82K+ Points: 4

Given a string **s** of length **n**, find all the **possible subsequences** of the string **s** in lexicographically-sorted order.

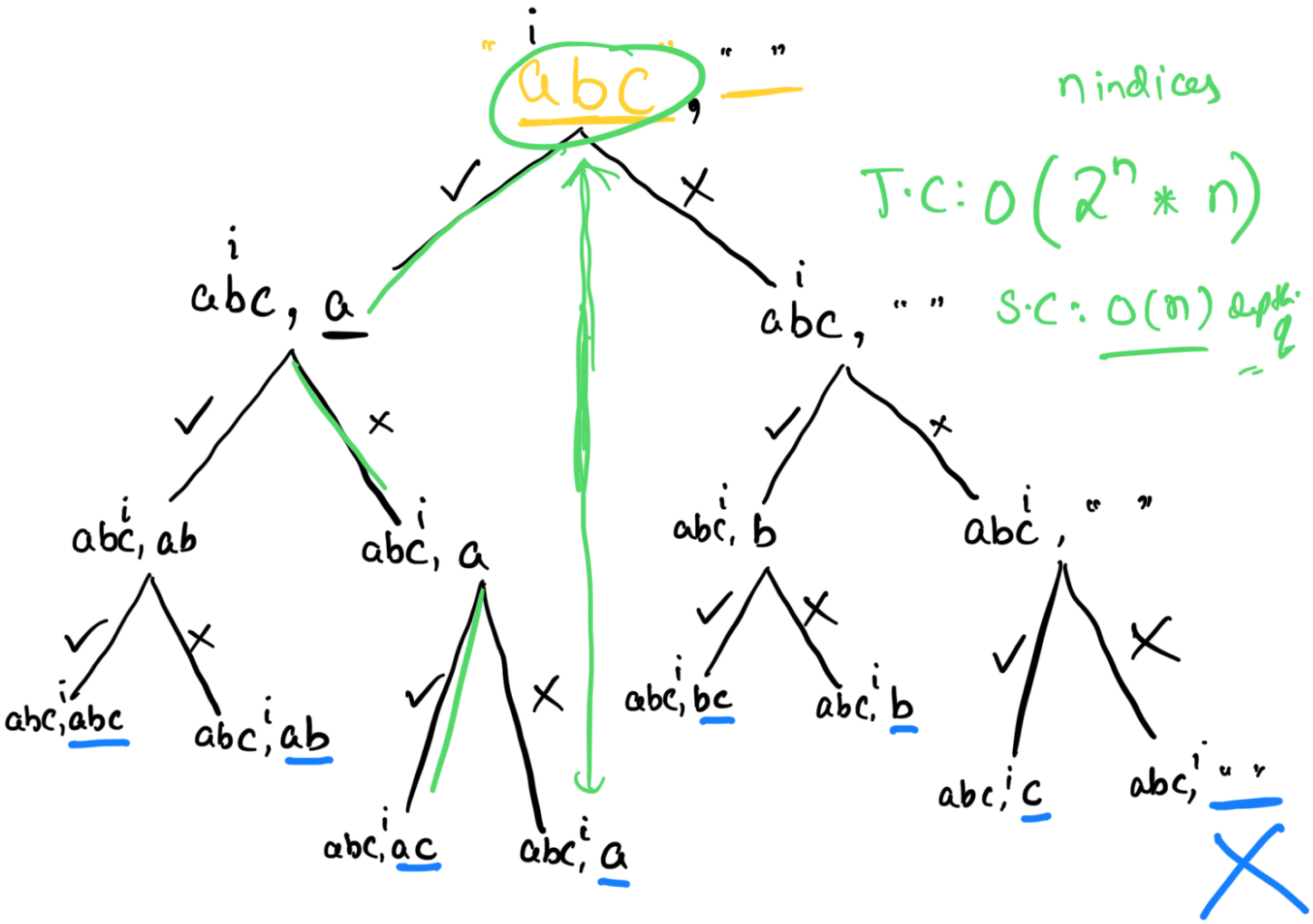
Example :- S = "abc"

Output = { "a", "ab", "abc", "ac", "b", "bc", "c" }

"abc" → {""}, {a}, {b}, {c}, {ab}, {ac}, {bc}, {b}

Recursion Tree :-

(Same as Subsets)



Solve (S , curr , 0) ;

Solve(s , curr, idx) {

```
if ( idx >= s.length() ) {
    if (curr != "") {
        result.push_back(curr); return;
    }
}

curr.push_back(s[idx]);
solve(s, curr, idx+1);
curr.pop_back();
solve(s, curr, idx+1);
}

sort(result.begin(), result.end());
```