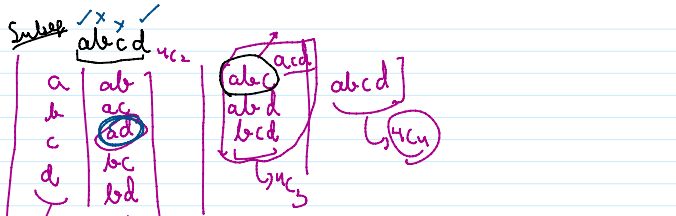
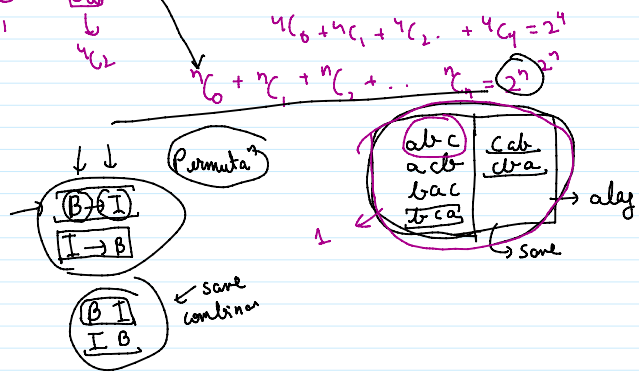
Subseq





n

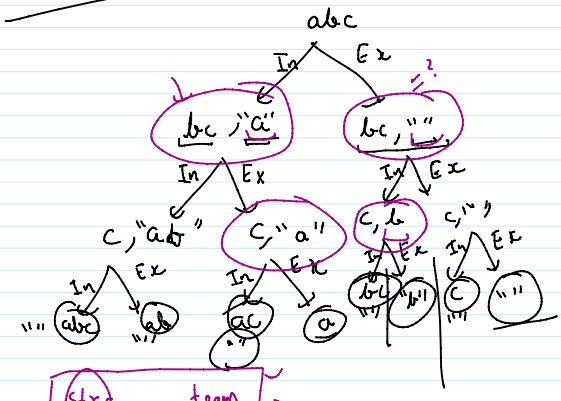
abcd

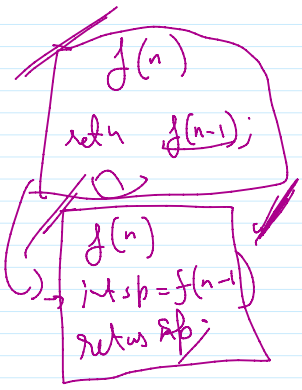
Subseq → seq Subst

Substring → (not subseq)

→ subset
→ seq
→ not

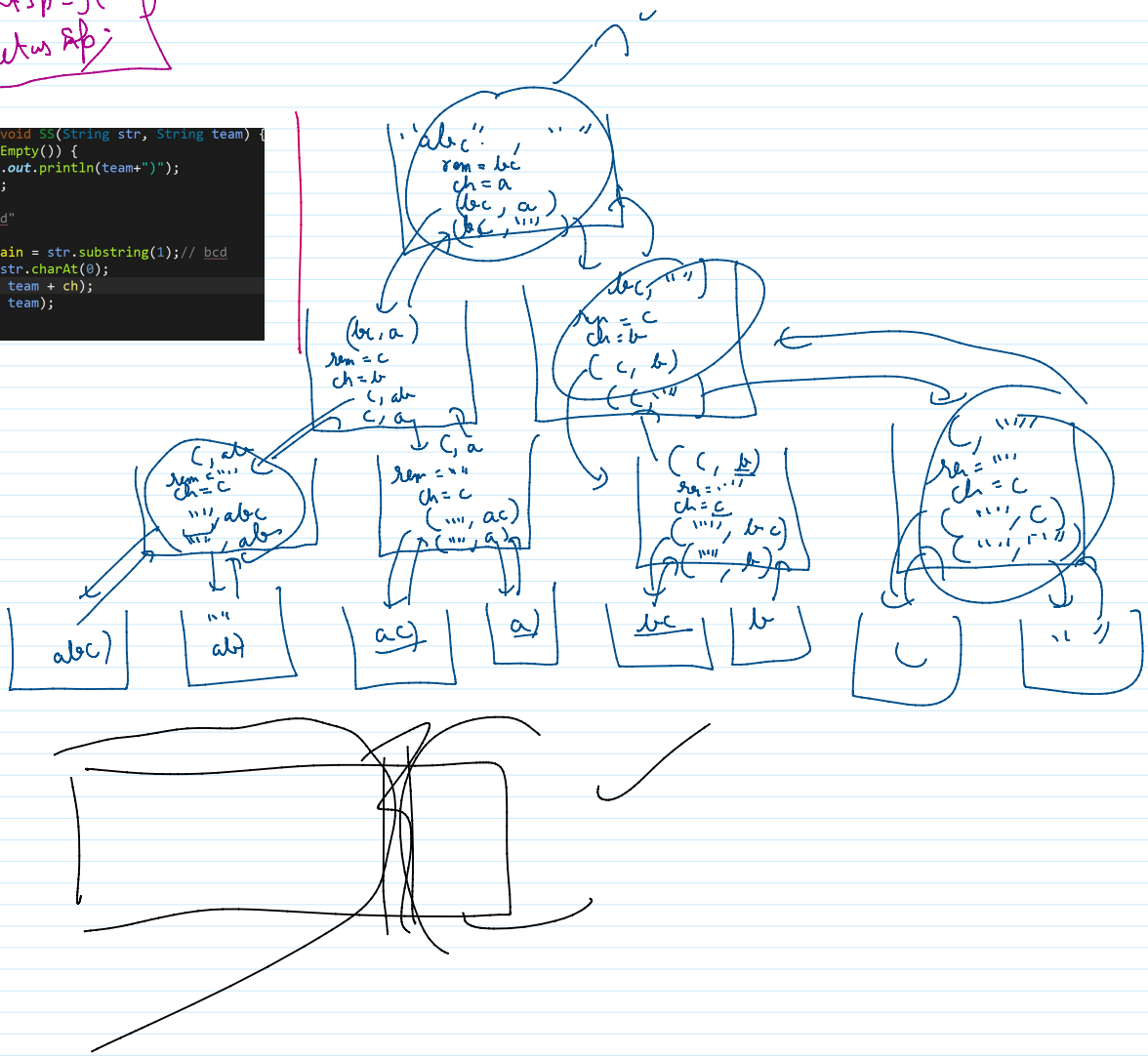
abcd
2·2·2·2
2^n

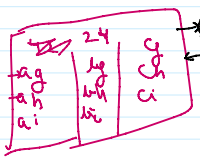


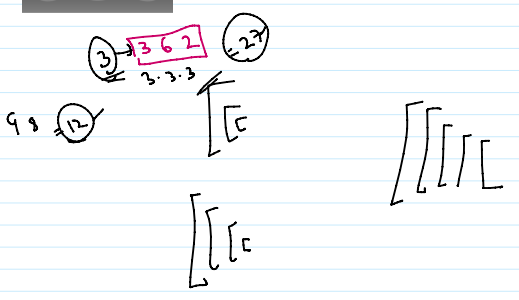


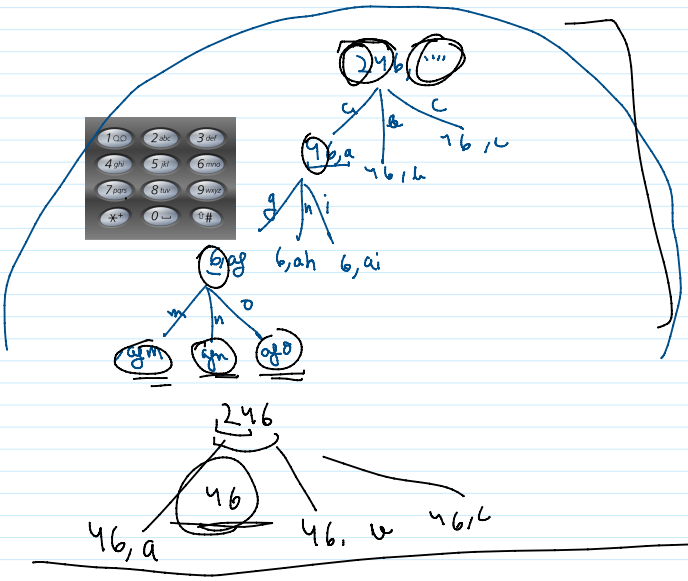

```java
public static void SS(String str, String team) {
    if (str.isEmpty()) {
        System.out.println(team+")");
        return;
    }
    str = "abcd"
    team = "Z"
    String remain = str.substring(1);// bcd
    char ch = str.charAt(0);
    SS(remain, team + ch);
    SS(remain, team);

}
```

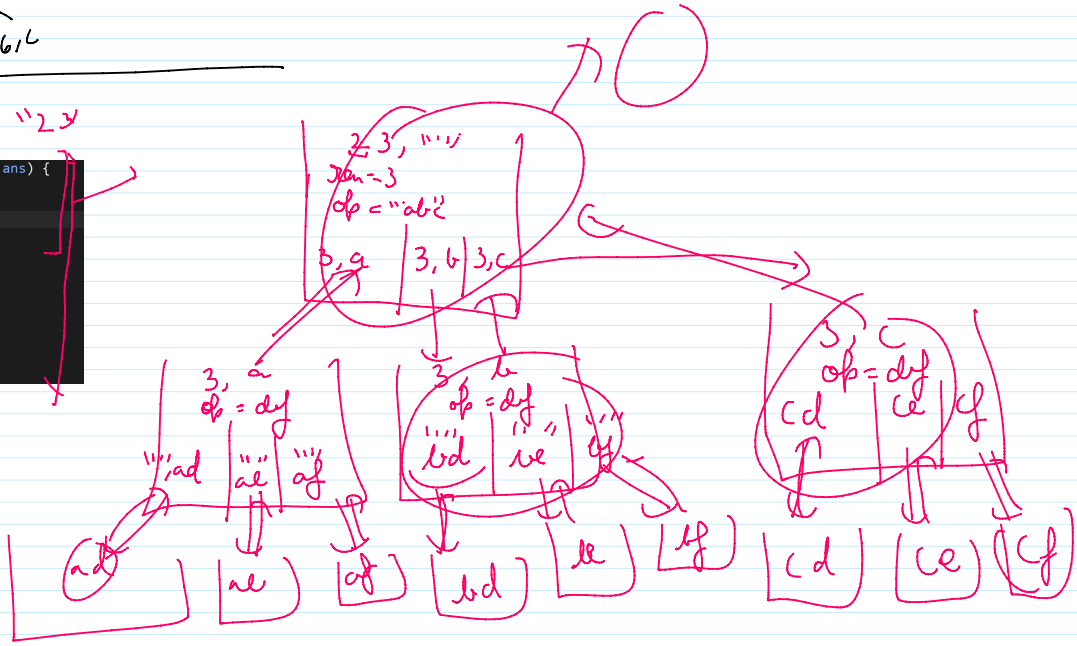




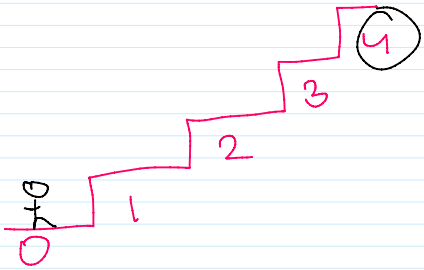


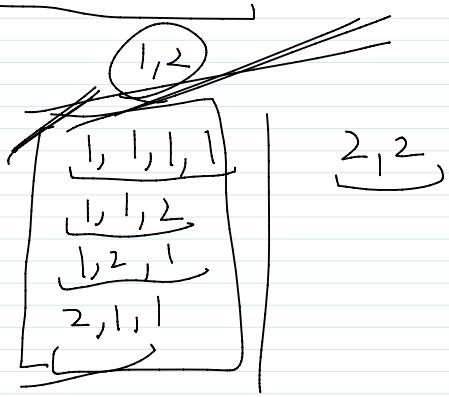



```
public static void solve(String buttons, String ans) {
    BP : buttons : 246
    if(buttons.isEmpty()) {
        System.out.println(ans);
        return;
    }
    String remain = buttons.substring(1);// 46
    char ch = buttons.charAt(0); //2
    String op = options(ch); //abc
    for(int i=0;i<op.length();i++) {
        solve(remain, ans+op.charAt(i));
    }
```



Climb stairs





Input: n = 3
Output: ["((()))","(()())","(())()","()(())","()()()"]