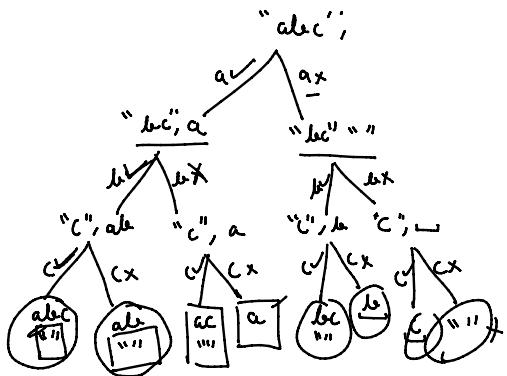
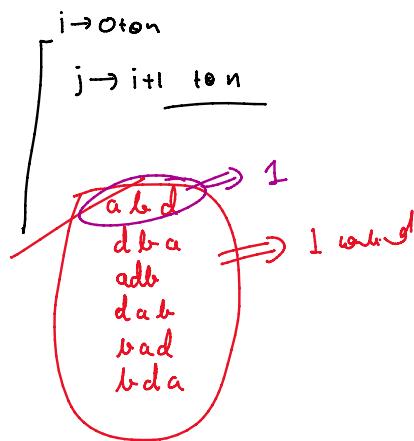
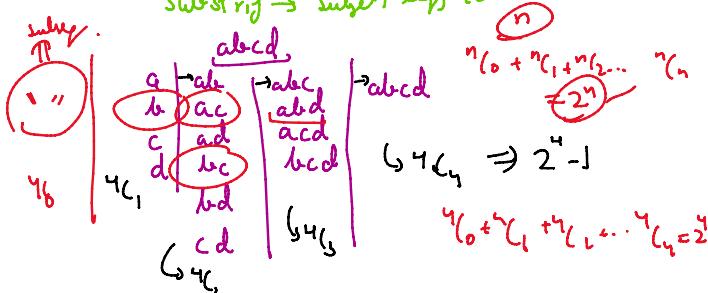


Subsequence → subset, seq.

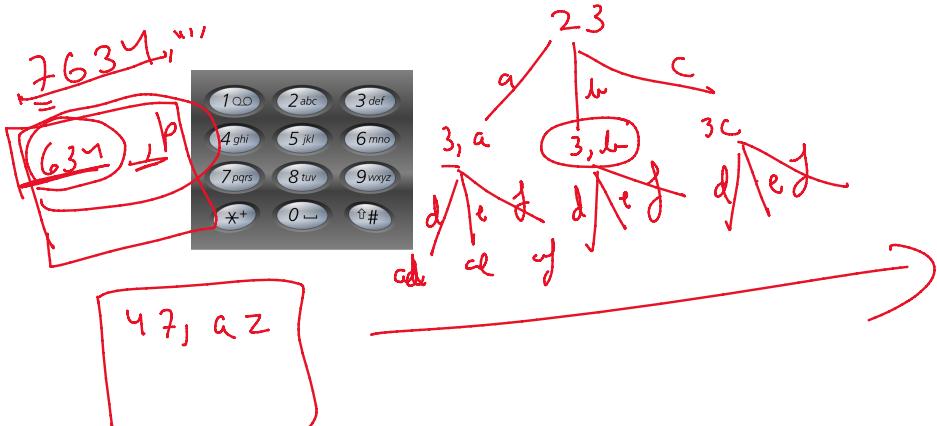
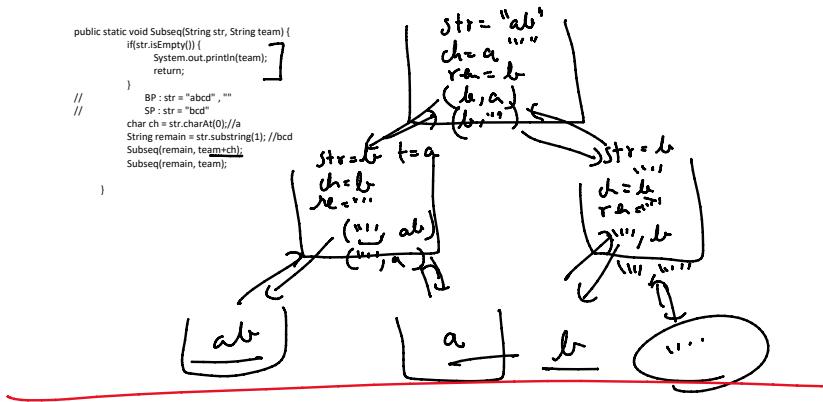
substring → subset, seq, cont.



```

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

```



$$\begin{aligned}
 '0' &\rightarrow x \\
 '1' &\rightarrow x+1 \\
 '2' &\rightarrow x+2 \\
 '5' &\rightarrow x+5
 \end{aligned}$$

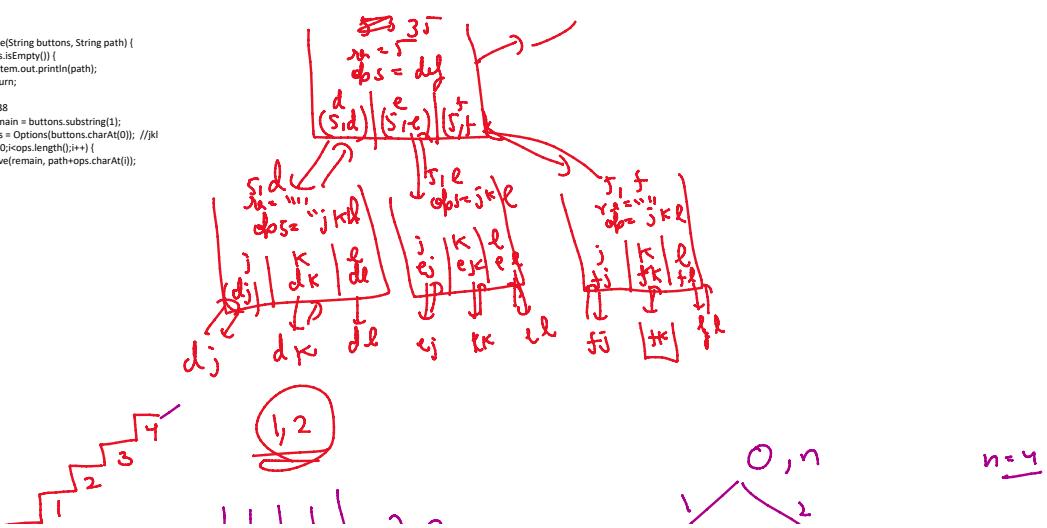
$'5' \rightarrow 5$

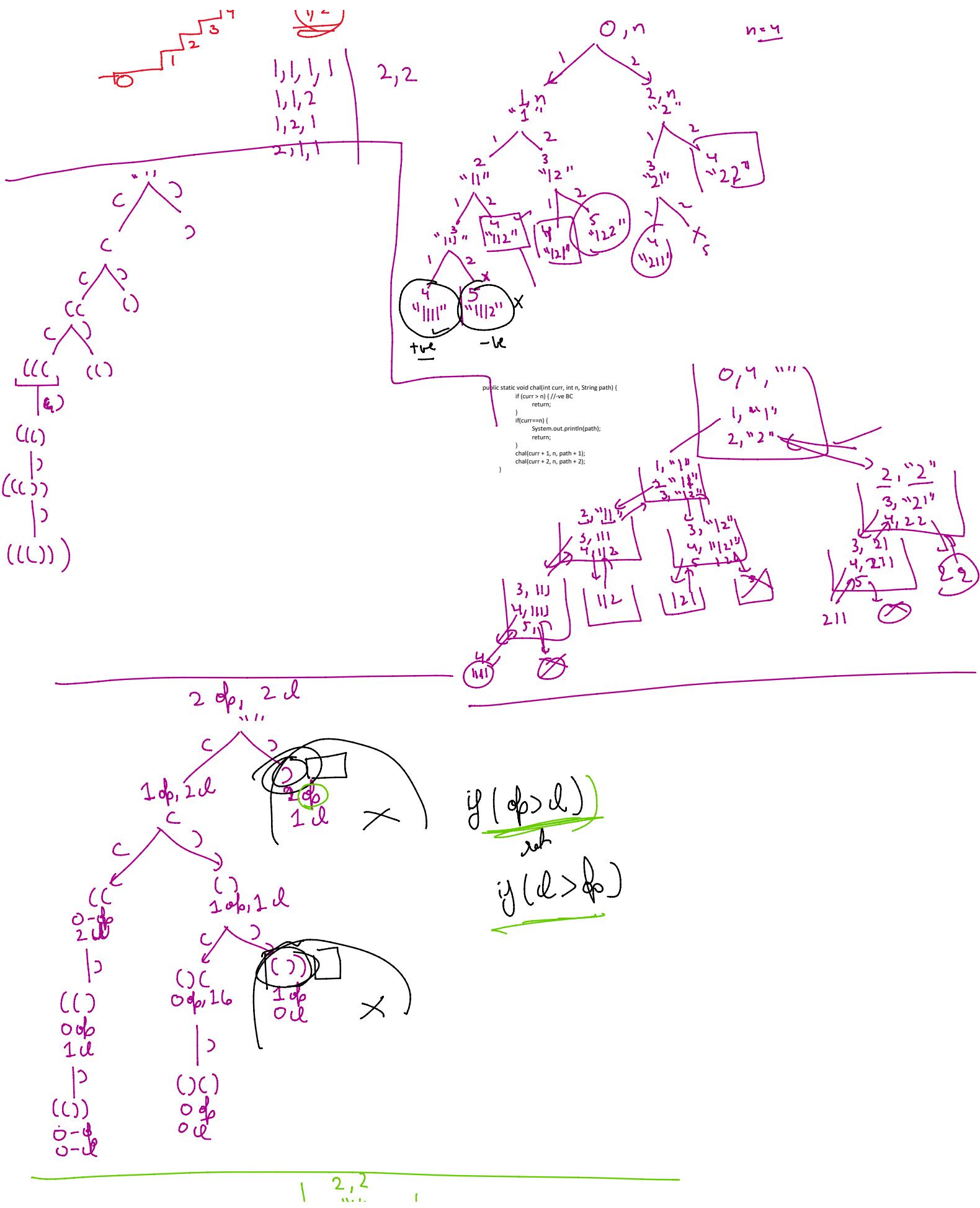
$$\boxed{'5' - \cancel{*} = 5}$$

```

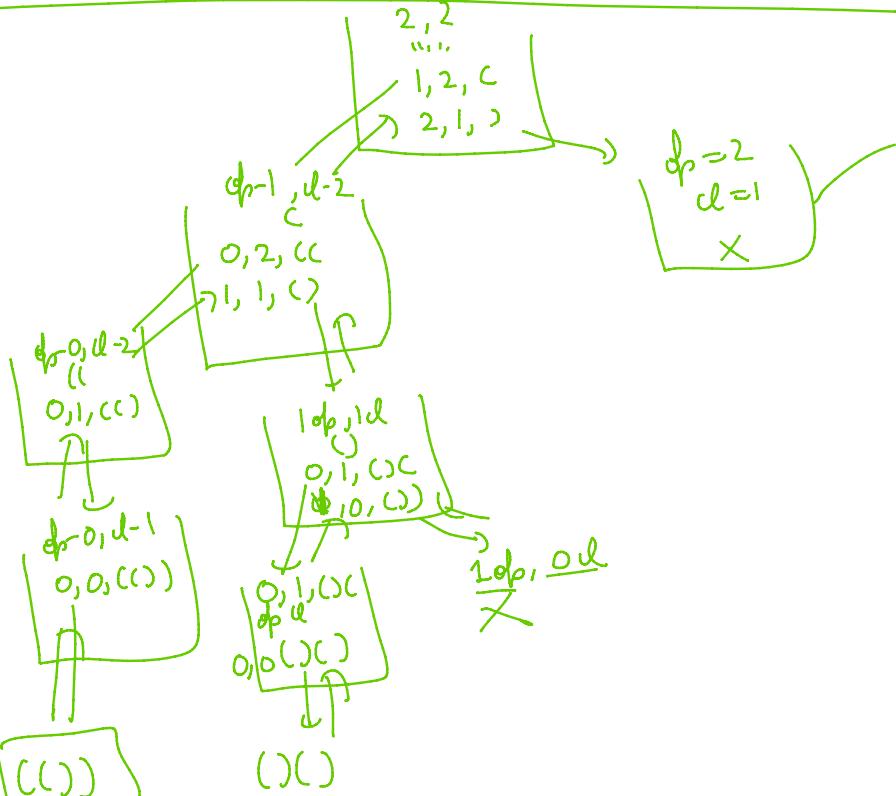
public static void solve(String buttons, String path) {
    if(buttons.isEmpty()) {
        System.out.println(path);
        return;
    }
    // buttons => 538
    String remain = buttons.substring(1);
    String ops = Options(buttons.charAt(0)); //jkl
    for(int i=0; i<ops.length(); i++) {
        solve(remain, path+ops.charAt(i));
    }
}

```

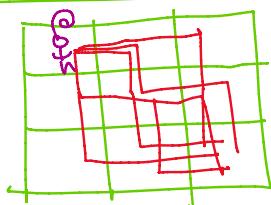




$O - d$

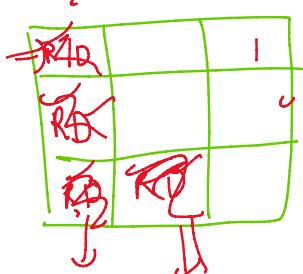


O, D
2, 2



$\overset{R}{\rightarrow}$
DJD
DRRD
DRDR
DDR

$O, 1$
 R
 $O, 0$



RRDD

R DRD

R DDR

D RRD

DR DR

DDR R



