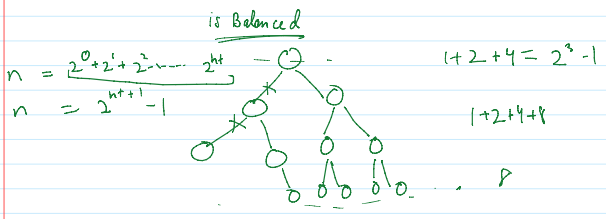
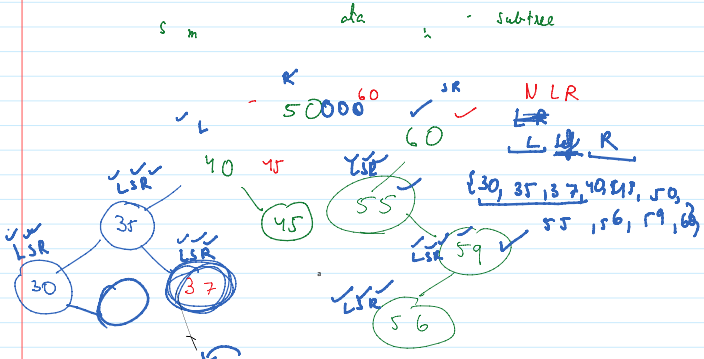
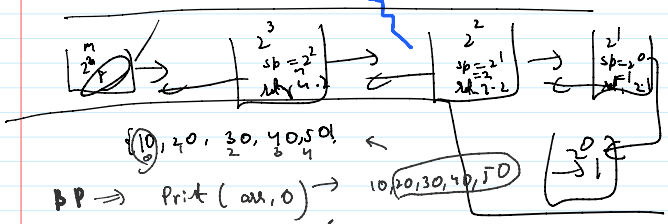
Saturday, September 18, 2021 10:59 AM



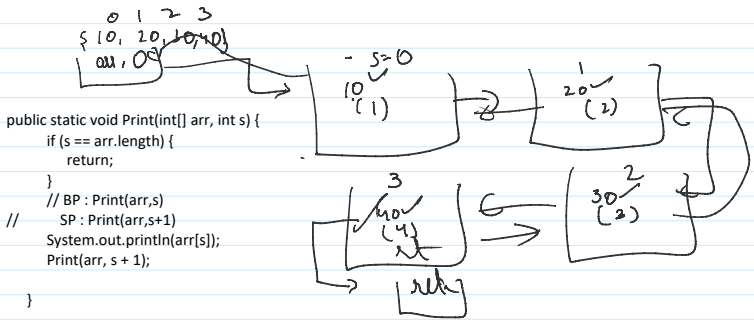


BST → Binary Search Tree



Biggest P → n
Smallest P → 1 (arr.length - 1)

S → arr.length - 1




```
public static void Print(int[] arr, int s) {
        if (s == arr.length) {
            return;
        }
        // BP : Print(arr,s)
//          SP : Print(arr,s+1)
        System.out.println(arr[s]);
        Print(arr, s + 1);

    }
```

0 1 2 3

```
}
```

{10, 20, 30, 40}
0 1 2 3

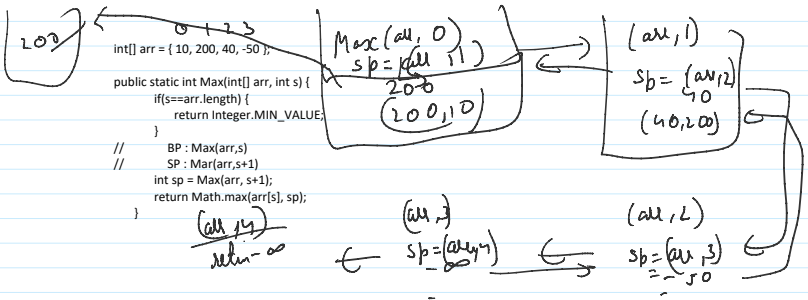
BP → Rev(0, arr) → 40, 30, 20, 10

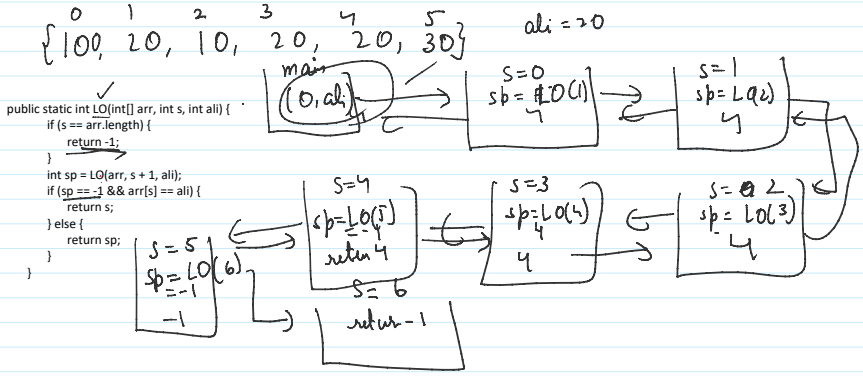
SP → Rev(1, arr) → 40, 30, 20

{100, 9, 8, 12, 5}
0 1 2 3 4

Max(arr, 0) → 12

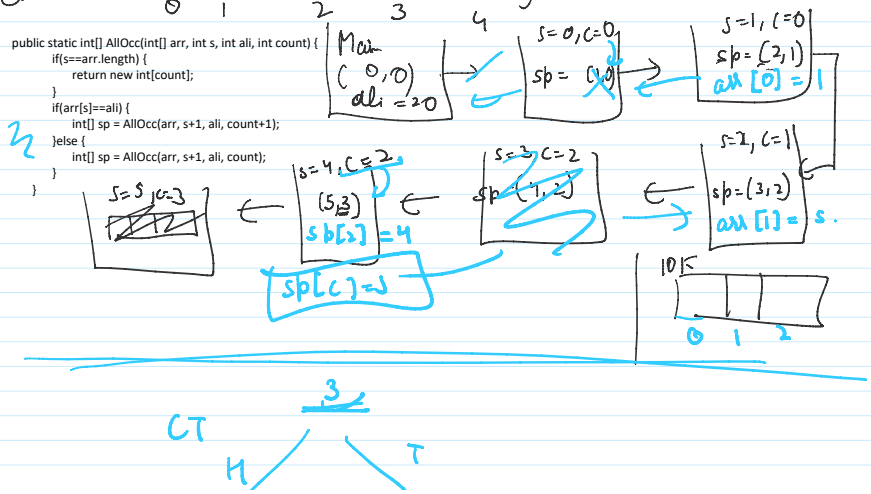
SP → Max(arr, 1) → 12

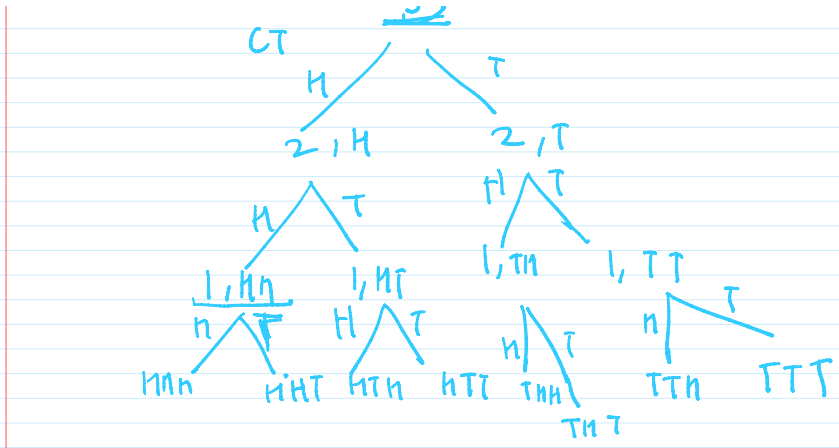




{10, 20, 2

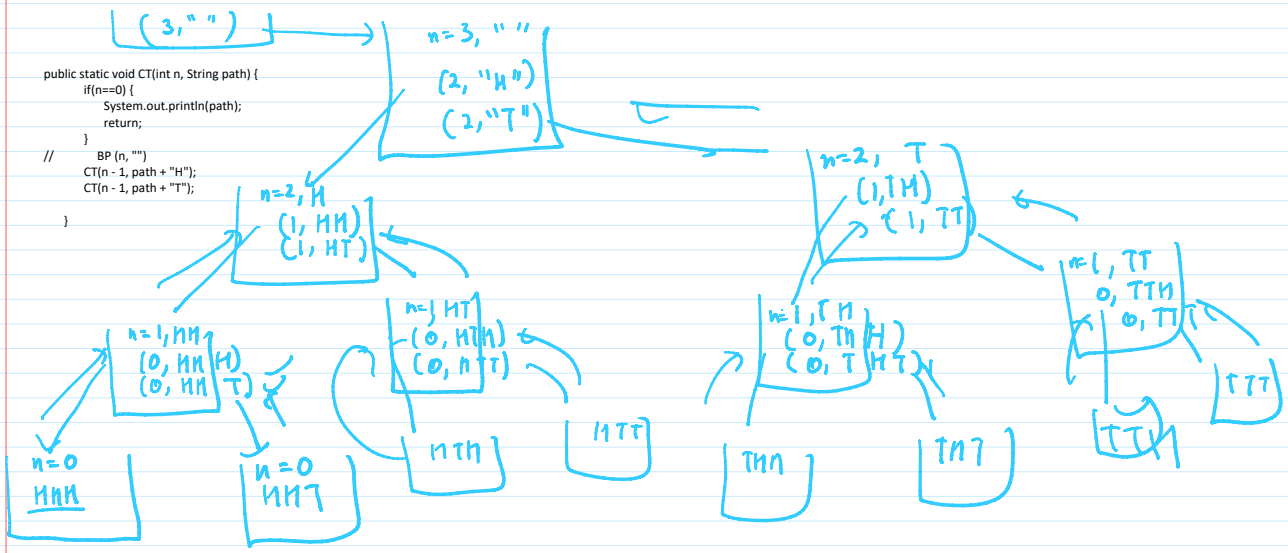
Q {10, 20, 20, 10, 20}
0 1 2 3 4

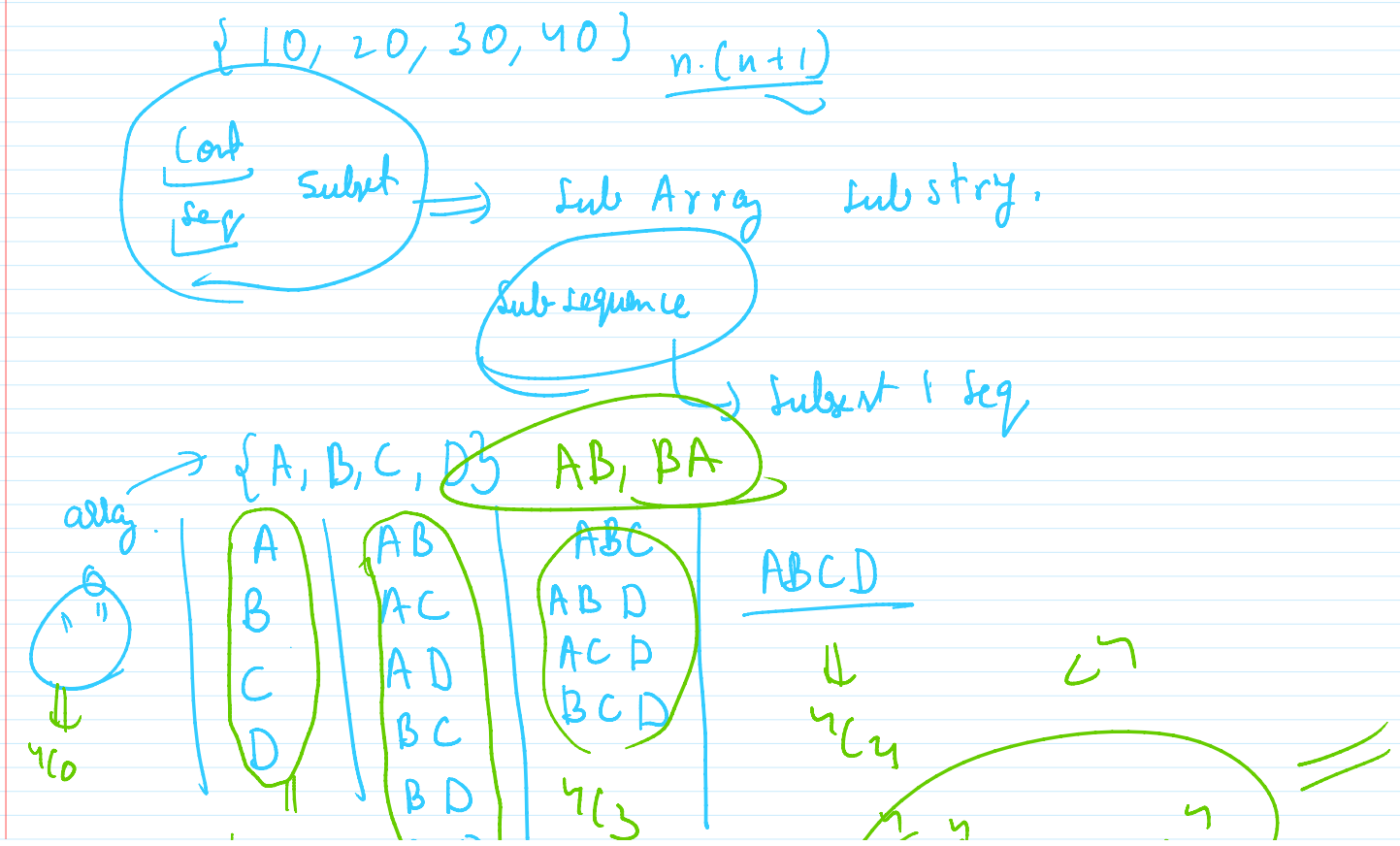





```
public static void CT(int n, String path) {
        if(n==0) {
            System.out.println(path);
            return;
        }
//          BP (n, "")
        CT(n - 1, path + "H");
        CT(n - 1, path + "T");

    }
```





$^4C_0$ | $^4C_1$ | BD, CD | $^4C_3$ | $^4C_4$

$$\sum_{r=0}^{4} {}^4C_r = 2^4$$

{ 10, 20, 30, 40 }

$^4C_2$

10,20 | 10,30 | 10,40 | 20,10 ✗ | 20,30 | 20,40

2

30,10 ✗

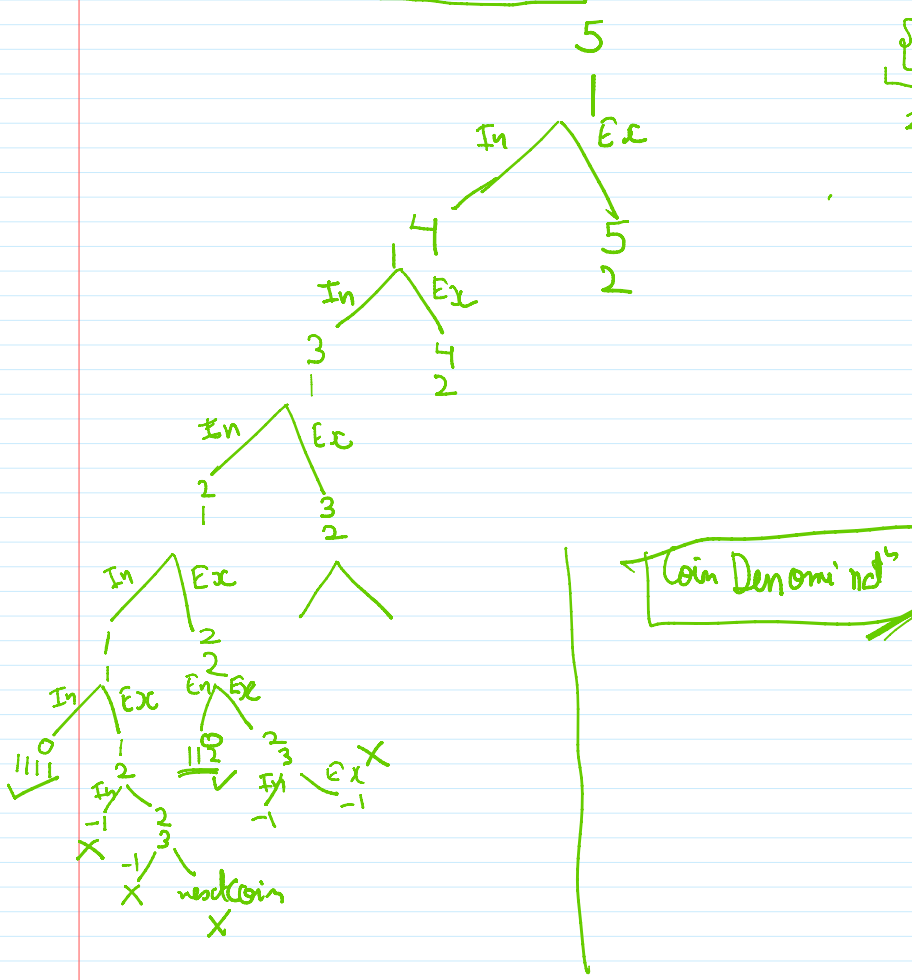
30,20 ✗

30,40 ✓

{1, 2, 3} → reach 5

{ }

2, 3, 4, 5



Coin Denomination → Live

0 1 2 3

{10, 20, 30, 40}

Max (0, arr) → 40

Max (1, arr)

sp = 40

return (40, arr[s])

}

{10, 20, 10, 20, 10}
0 1 2 3 4

0	2	4

(s, arr, ali, count) {

if (arr[s] == ali) {

int[] sp = AO(s+1, ali, c+1)

} else

int[] sp = (s+1, c) ×

if (s == arr.length)

return new int[c]

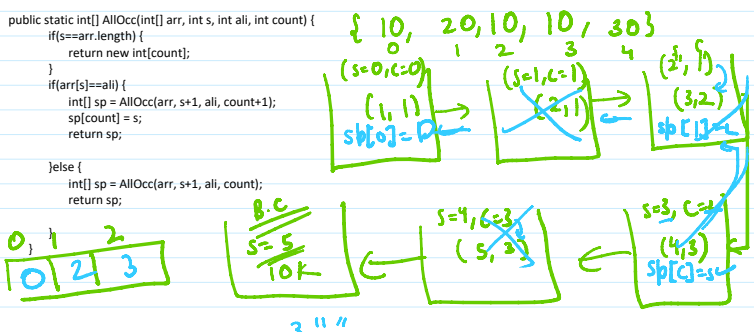
{ _ _ _ _ , -100 }
3

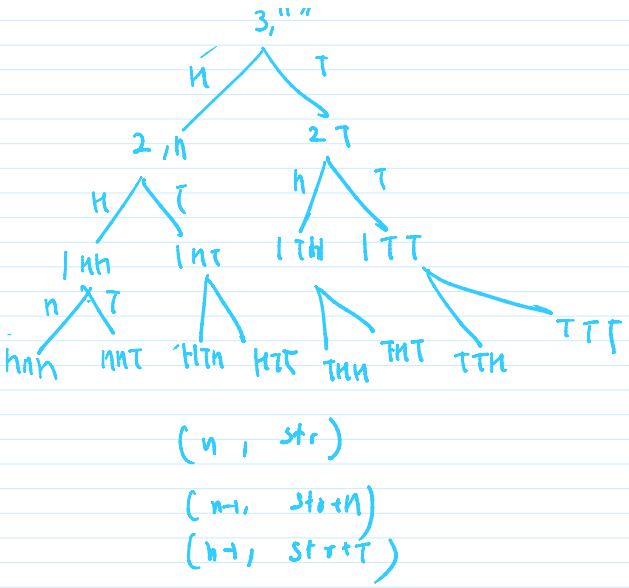
-100 ⇒ Max(s, -100)

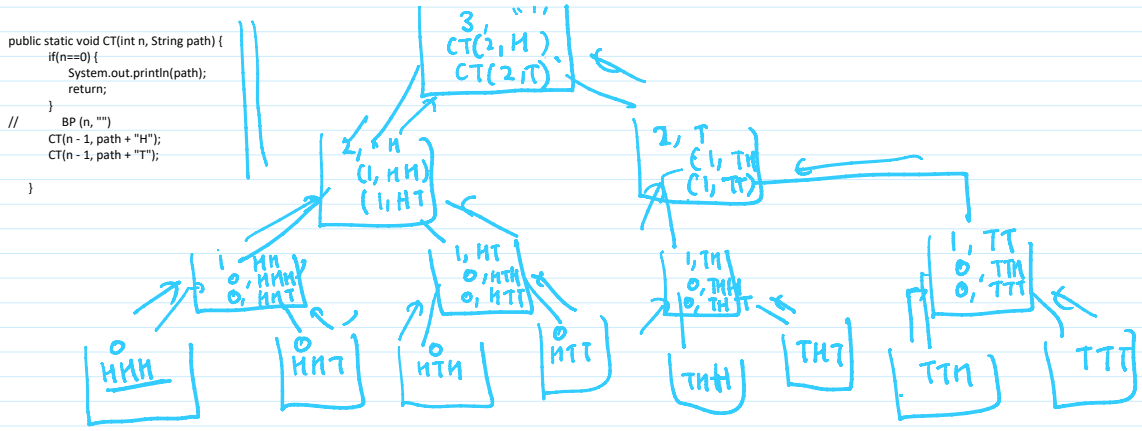
x ⇒ Max(res.sp, x)

```
public static int[] AllOcc(int[] arr, int s, int ali, int count) {
    if(s==arr.length) {
        return new int[count];
    }
    if(arr[s]==ali) {
        int[] sp = AllOcc(arr, s+1, ali, count+1);
        sp[count] = s;
        return sp;

    }else {
        int[] sp = AllOcc(arr, s+1, ali, count);
        return sp;
    }
```







2 2 2

Sub sequence

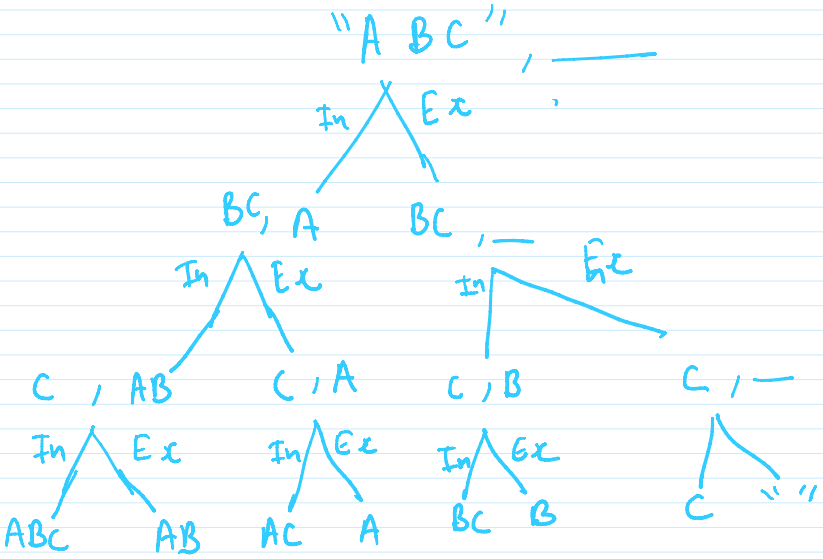
Subset + Seq

$\sum_{r=0}^{n} {}^{n}C_r \Rightarrow 2^n$

$^nC_0$

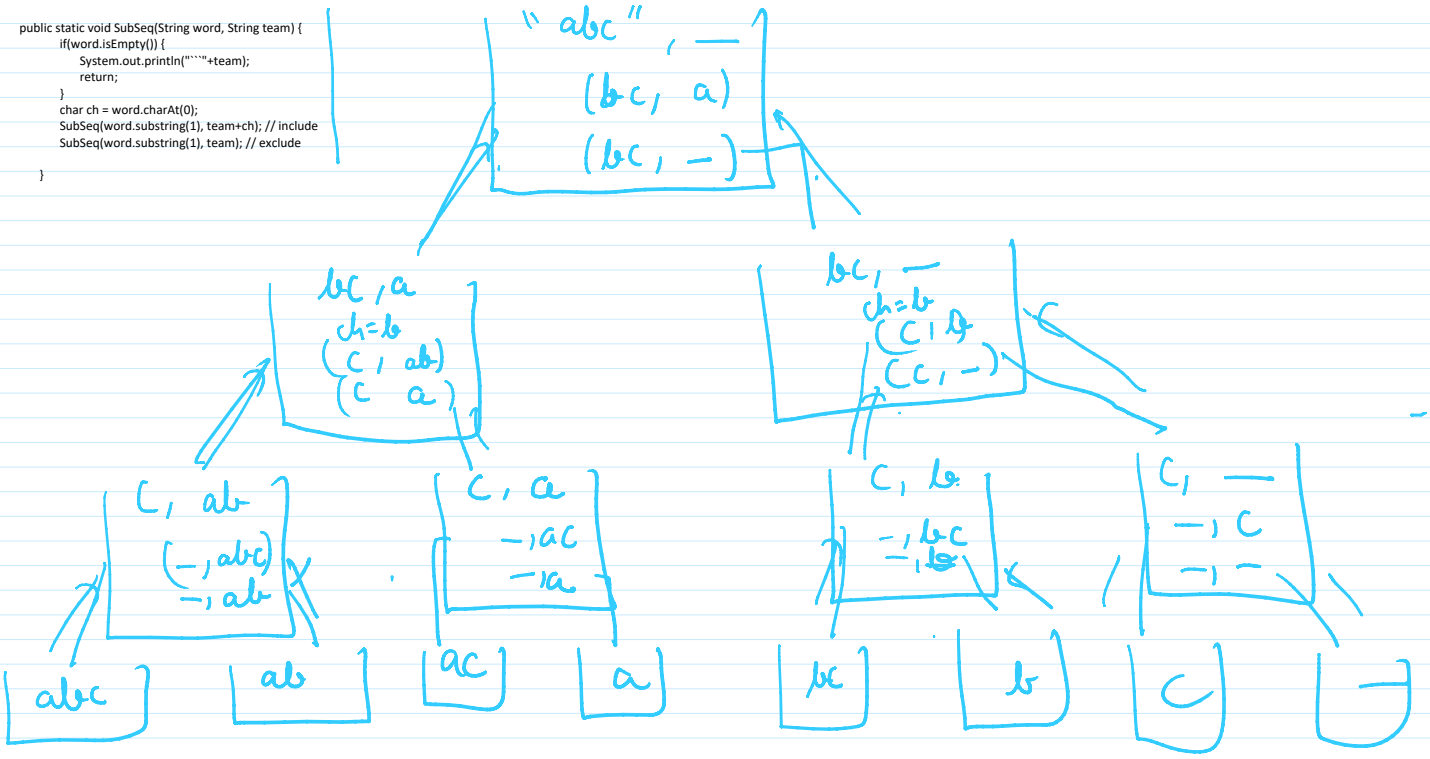
abcd

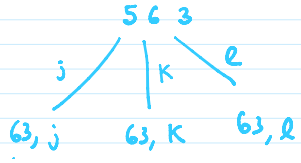
$^nC_1$	$^nC_2$	$^nC_3$	$^nC_4$
a	ab	abc	abcd
b	ac	abd	
c	ad	acd	
d	bc	bcd	
	bd		
	cd		

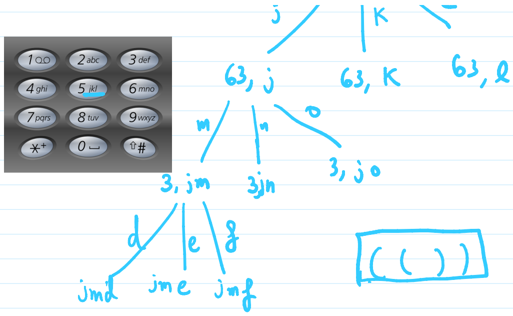


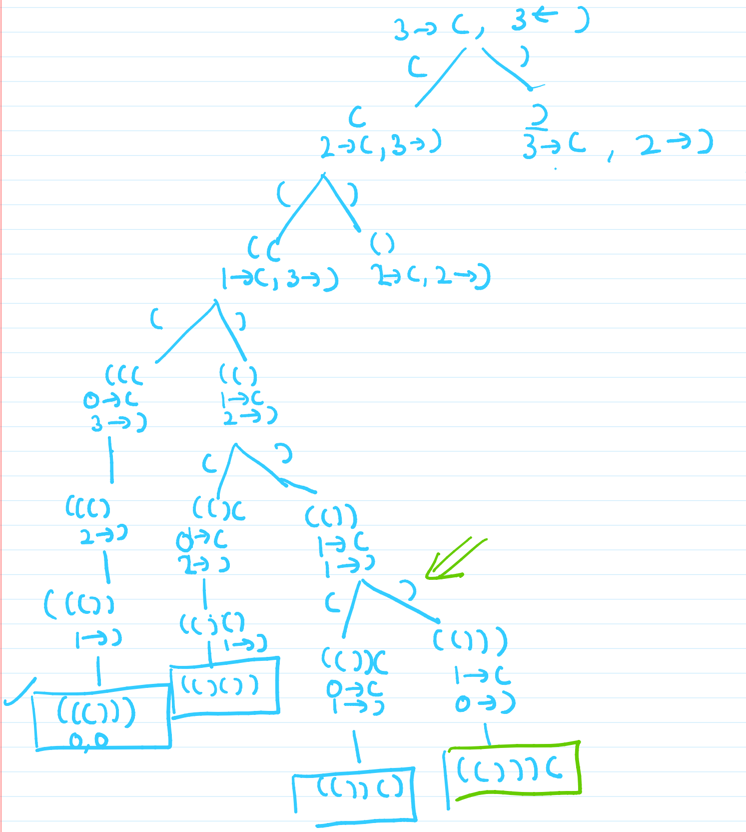

```
public static void SubSeq(String word, String team) {
        if(word.isEmpty()) {
            System.out.println("```"+team);
            return;
        }
        char ch = word.charAt(0);
        SubSeq(word.substring(1), team+ch); // include
        SubSeq(word.substring(1), team); // exclude

    }
```






1
2 abc
3 def
4 ghi
5 jkl
6 mno
7 pqrs
8 tuv
9 wxyz
0
#
63, j
63, K
63, l
3, jm
3jn
3, jo
jmd
jme
jmf
(())


3→(, 3→)
2→(, 3→)
3→(, 2→)
1→(, 3→)
2→(, 2→)
((())
((()))
(()())
(())()
(()))(