

# Ques Subsequences of a string

abc  $\Rightarrow$  "" ab  
a ac  
b bc abc  
c

$2^3 \Rightarrow 8$

→  
- - -  
a - -  
- b -  
- - c  
a b -  
a - c  
- b c  
a b c

ab  $\Rightarrow$  - -  
a -  
- b  
a b

$2^2 = 4$

abc

bc  $\Rightarrow$  - - -  
- b -  
- - c  
- b c  
a - -  
a b -  
a - c  
a b c

## Expectation

ss(abc)

→  
- - -  
a - -  
- b -  
- - c  
a b -  
a - c  
- b c  
a b c

## Faith

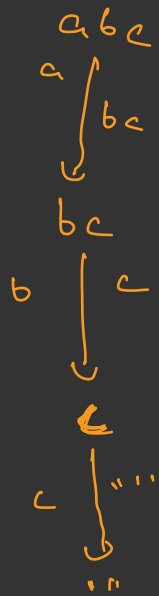
ss(bc)

- -  
b -  
- c  
b c

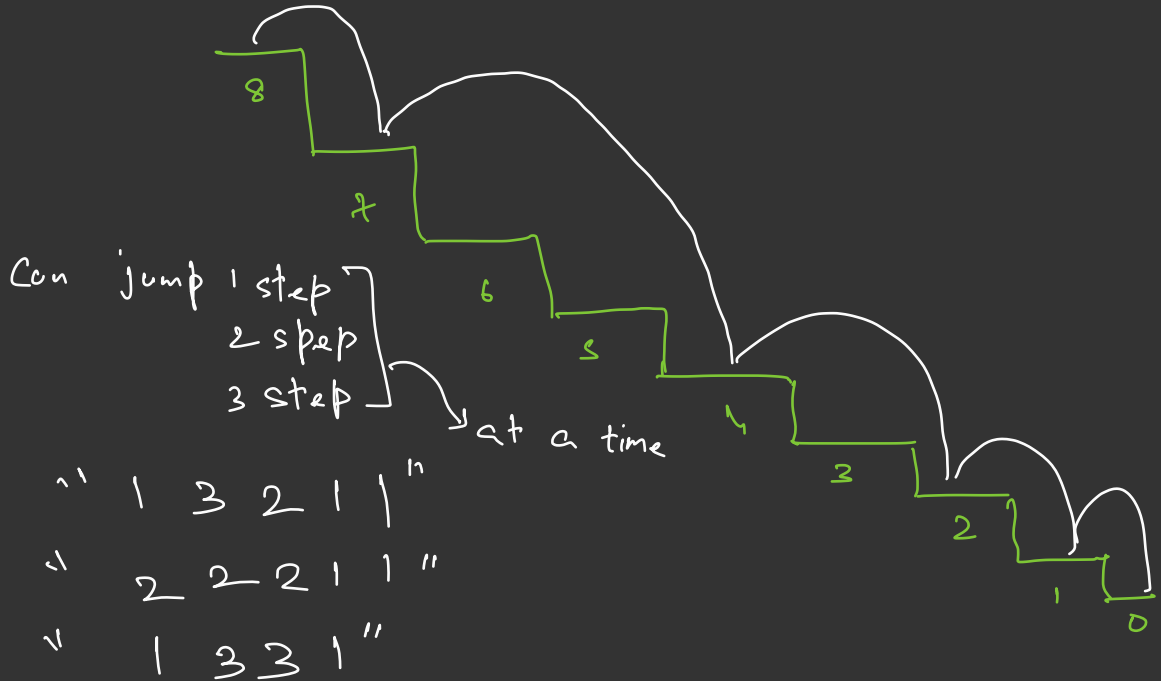
## Combine

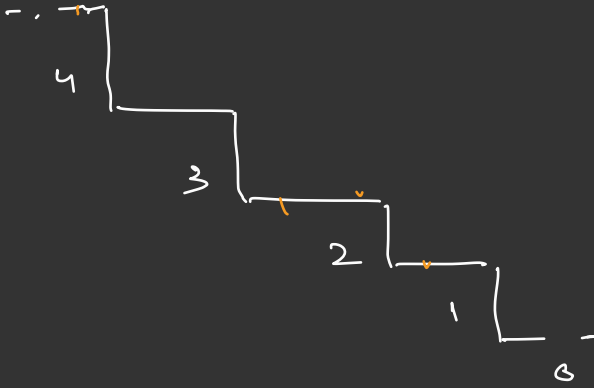
AL  $l = ss(bc)$ ;  
new AL res  
res.add(l + "")  
res.add(a + l)

```
public static ArrayList<String> subSequence(String s) {  
    if(s.length() == 0) {  
        ArrayList<String> res = new ArrayList<>();  
        res.add("");  
        return res;  
    }  
  
    char c = s.charAt(0);  
    String rem = s.substring(1);  
  
    ArrayList<String> recResults = subSequence(rem);  
    ArrayList<String> myResults = new ArrayList<>();  
  
    for(String ss : recResults) {  
        myResults.add("" + c + ss);  
    }  
  
    for(String ss : recResults) {  
        myResults.add(ss);  
    }  
  
    return myResults;  
}
```

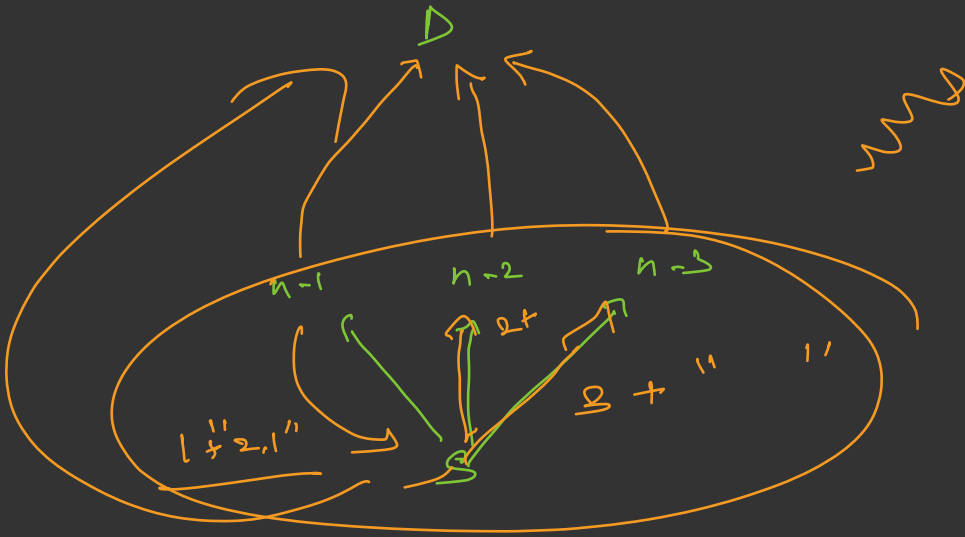


## Ques Stair Path





1	1	1	1
1	2	1	
2	1	1	
3	1		



Expectation

$SP(4, 0)$

$\rightarrow$



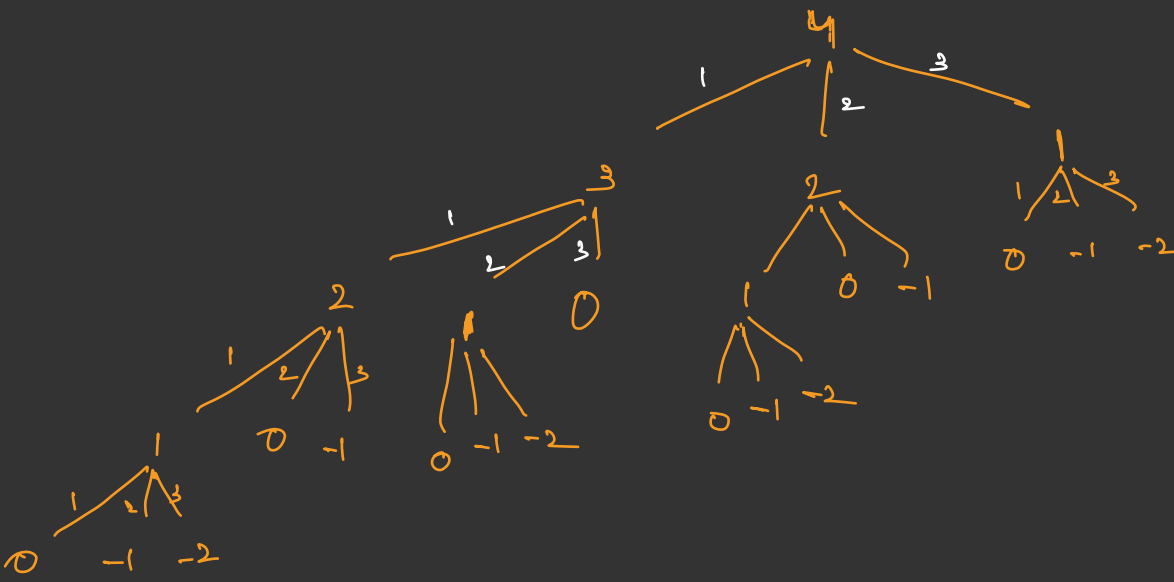
Faith

$SP(3, 0)$

$SP(2, 0)$

$SP(1, 0)$

Combination



get  $\rightarrow$  AL < s >  $\rightarrow$  all the paths

print SP  $\rightarrow$  print all paths

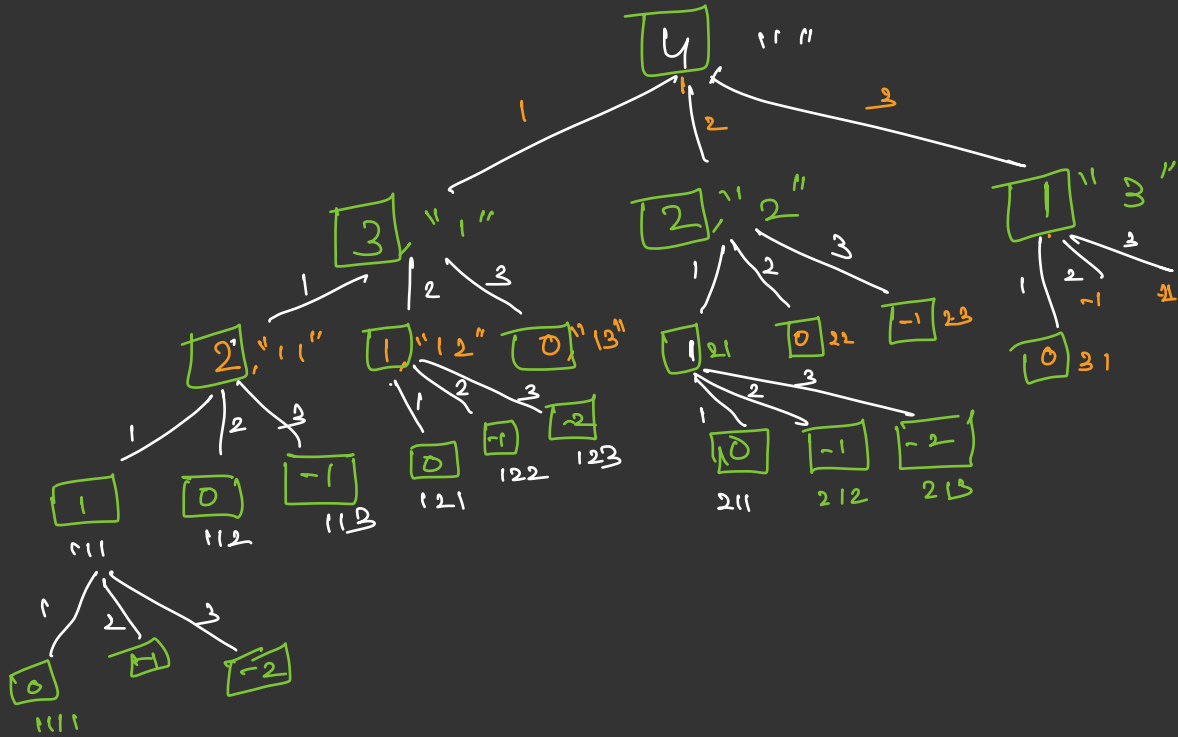
psg



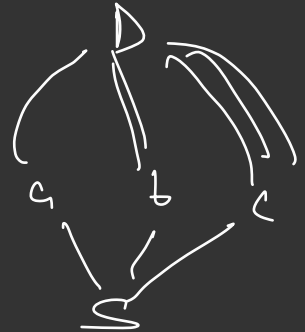
```

public static void printStairPaths(int n, String psf) {
    if(n == 0) {
        System.out.println(psf);
        return;
    } ,
    if(n < 0) return;
    printStairPaths(n - 1, psf + 1); ←
    printStairPaths(n - 2, psf + 2); ←
    printStairPaths(n - 3, psf + 3); ←
}

```



Queue Count Stairpath



Count

Ques

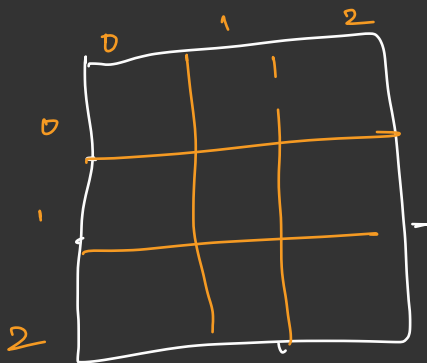
Print maze path

How

Ques Get maze path

R

D



R R D D

R D R D

R D D R

D R R D

D R D R

D D R R

$(0,0) \rightarrow (2,2)$

$(2,2)$  D

R D D

D R D

D D R

$(0,1)$

$(1,0)$

R

D

$(0,0)$  S

R R D

R D R

D R R

Expectation

GMP  
(area, 0, 0)

Fcith

GMP(area, 0, 1)

GMP(area, 1, 0)

Combine

R+

D+

---

Que Count Maze Path

