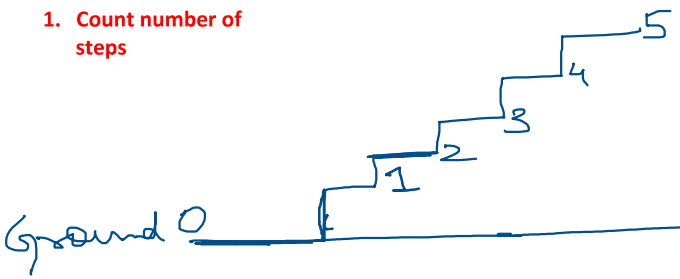
June 25, DP

25 June 2020 21:20

1. **Count number of steps**

I will come from level 1 or level 0



$\boxed{1 \text{ way} \rightarrow 1 \text{ way}}$

$\boxed{0 + 2 \text{ steps} \rightarrow 1 \text{ way}}$

$\boxed{\text{To reach } 3}$

$0 + 3 \text{ steps} \rightarrow$ Jump 0 to the

$1 + 2 \text{ steps} \rightarrow$ Jump 1 take palindrome to the

$2 + 1 \text{ step} \rightarrow$ Jump 2 take palindrome to the

Level 1 $\rightarrow$ 1

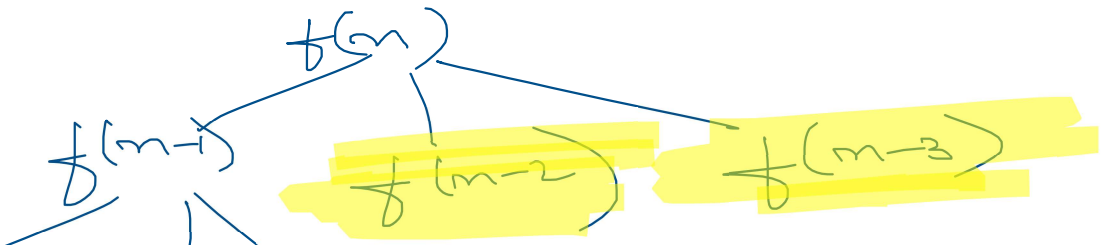
Level 2 $\rightarrow$ 1+1, 2 $\rightarrow$ 2

Level 3 $\rightarrow$ 1+1+1, 1+2, 2+1, 3

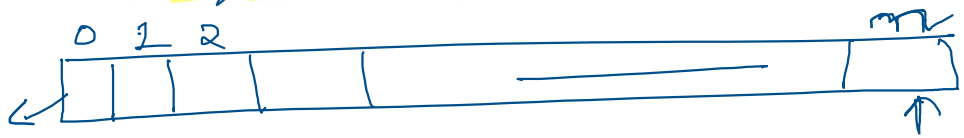
$$\boxed{\text{FindStep}(3) = \text{FindStep}(4) + \text{FindStep}(3) + \text{FindStep}(2)}$$

$$\boxed{\text{FindStep}(n) = \text{FindStep}(n-1) + \text{FS}(n-2) + \text{FS}(n-3)}$$

$\boxed{\text{State Expression}}$



f(n-2) f(n-3) f(n-4)



Min no of ways to reach 0

Min no. of ways to reach n

2. Edit Distance

S1(m)	S2(n)	Result
—	—	0
—	ab	2
ab	—	2
a	a	

insert
Replace
Remove

Base Condition →

if (m==0) return n;
if (n==0) return m;

edit(String S1, String S2, int m, int n)
{ → Base Cons
if (charAt(m-1) == charAt(n-1))
{ edit(S1, S2, m-1, n-1);
}

S1 a
S2 a

S1 a b m
S2 a b n

Replacement }

a b d m
a b d n

1 + edit(S1, S2, m-1, n-1)

ab c

Insertion

ab cd
ab d

$1 + edit(s1, s2, m, n-1)$

Minimum of these three

Removal

ab ~~c~~
ab d

$1 + edit(s1, s2, m-1, n)$

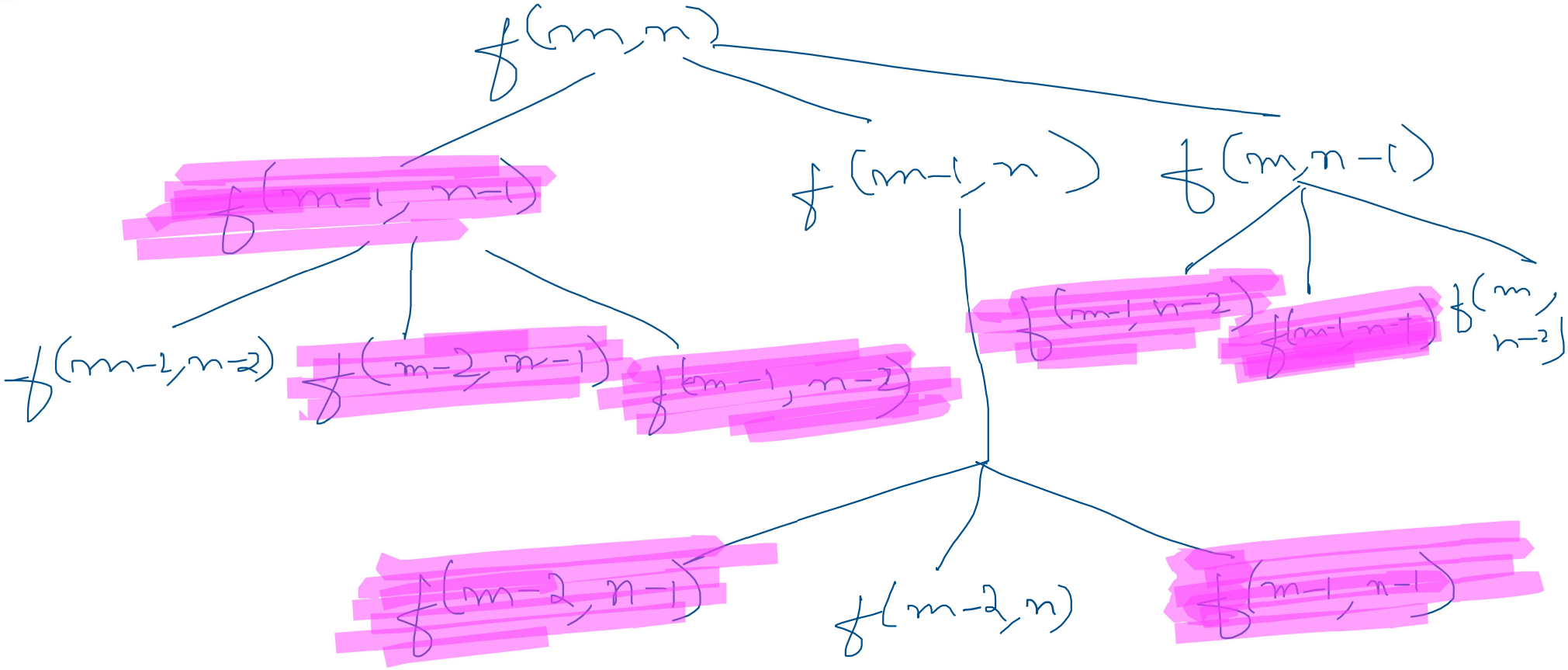
Or

$1 + Min(edit(s1, s2, m-1, n-1),$ → Replacement

$edit(s1, s2, m-1, n),$ → Removal

$edit(s1, s2, m, n-1);$ → Insertion

Recursion Tree



Overlapping SubProblems ✓

Optimal SubStructure ✓

Memoize

1 → Array (m, n) e.g. → $(2,2)$ → index

$(m+1) \times (n+1)$

$(2,2) \to (2,1)\ (2,0)$

$(1,2) \to (1,1) \to (1,0)$

$(0,2) \to (0,1) \to (0,0)$

	0	1	2	3	4
0					
1			(i,j)		
2					
3					$(m-1,n)$
4				$(m,n-1)$	(m,n)

→ At each place we will store the min. edit Distance to convert S1 with length i to S2 with length j

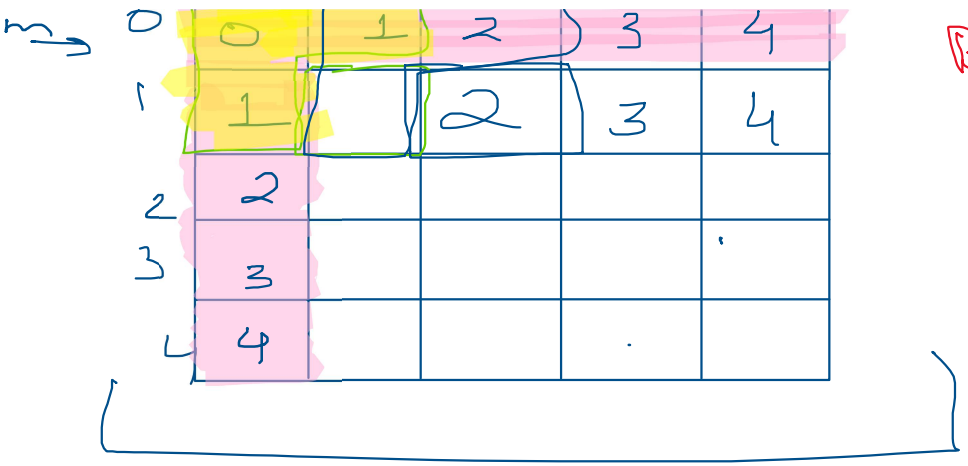
Tabulation

m

a

b

0	1	2	3	4



Base Condition →

```
if (m==0) return n;
if (n==0) return m;
```