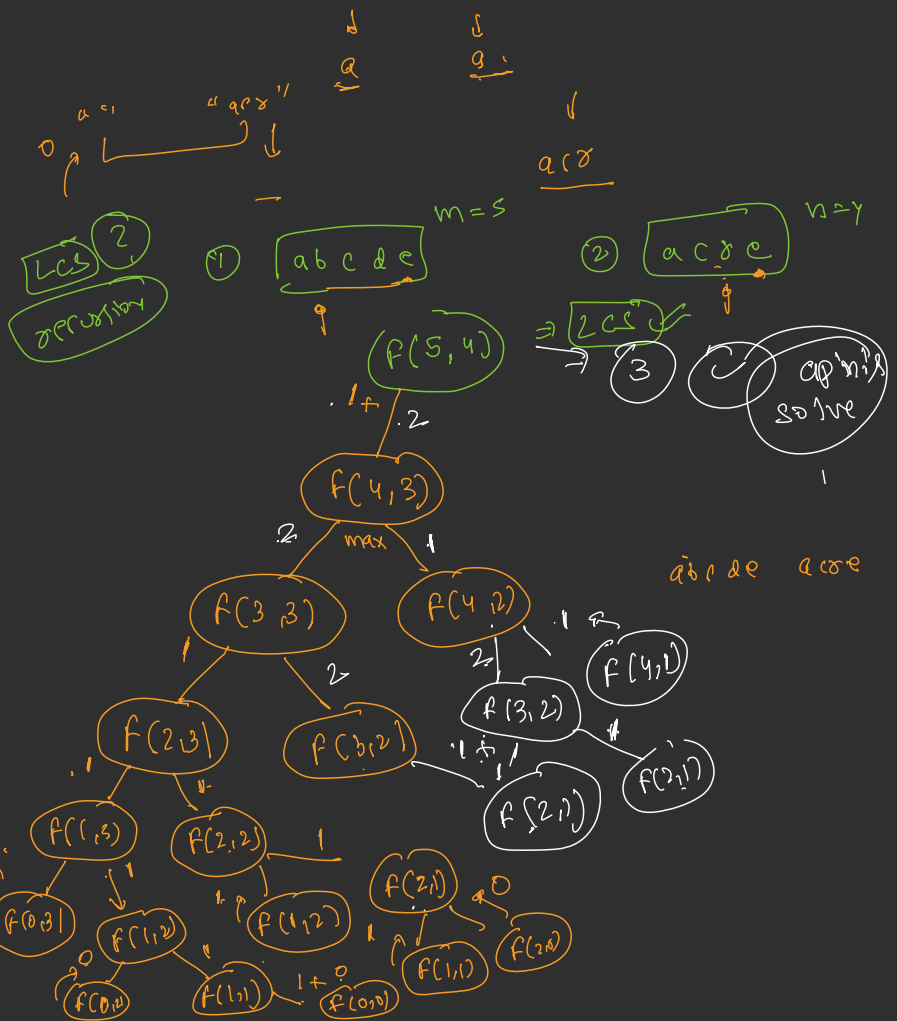

a
a
"ac"
"acr"
acr
LCS
2
operation
① abcde
m=5
② acre
n=4
f(5,4)
LCS
3
optimal solve
f(4,3)
max
f(3,3)
f(4,2)
f(2,3)
f(3,2)
f(1,3)
f(2,2)
f(0,3)
f(1,2)
f(1,1)
f(0,2)
f(0,1)
abcde acre

(2) a b c d

(2) d x a o

(1) ✓

a d

a , b a

b a

a d , b

a b c d e

a b c d e , f a b c e

f a b c d

a b c d , f a b c d

```
if ( text1[i-1] == text2[j-1] )
    dp[i][j] = 1 + dp[i-1][j-1]
else
    dp[i][j] = max(dp[i-1][j], dp[i][j-1])
```

a , a c

① "abcde" ② "acre" "ace"

	0	1	2	3	4
0	0	0	0	0	0
1 a	0	1	1	1	1
2 b	0	1	1	1	1
3 c	0	1	2	2	2
4 d	0	1	2	2	2
5 e	0	1	2	2	3

answer

i = m , j = n

result

ece a ace

```
while (i > 0 && j > 0)
{
    if (text1[i-1] == text2[j-1])
    {
        result += text[i-1];
        i--; j--;
    }
    else
    {
        if (dp[i-1][j] > dp[i][j-1])
            i--;
        else
            j--;
    }
```

len 5

abcebda

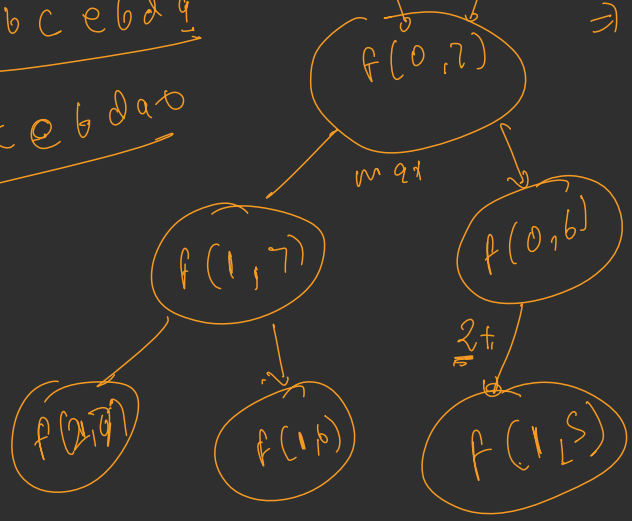
bcebda

① ⇒ ①

"a b c e b d a"

0 1 2 3 4 5 6 7

start, end



⇒ Base condition

```
if (start == end)
    return 1
if (start > end)
    return 0;
}
```

a a a a

$2+2+2+2$

$\downarrow$

8

"m b a d m" ← string

ins (2)

m b a d m ⇒ (3)

m b d a d b m

a r o a ⇒ LBS (3) (1)

a o r o a

(1) "aa" (2) "aa"

add del

(1)

(2) +3 ⇒ (5) add

(1) "abcd" "rhakc"

(2)

rahdK