

## Wildcard Matching

Input:

Output: True

$S = \text{“ca} * \text{bp?aab}\alpha\text{”}$

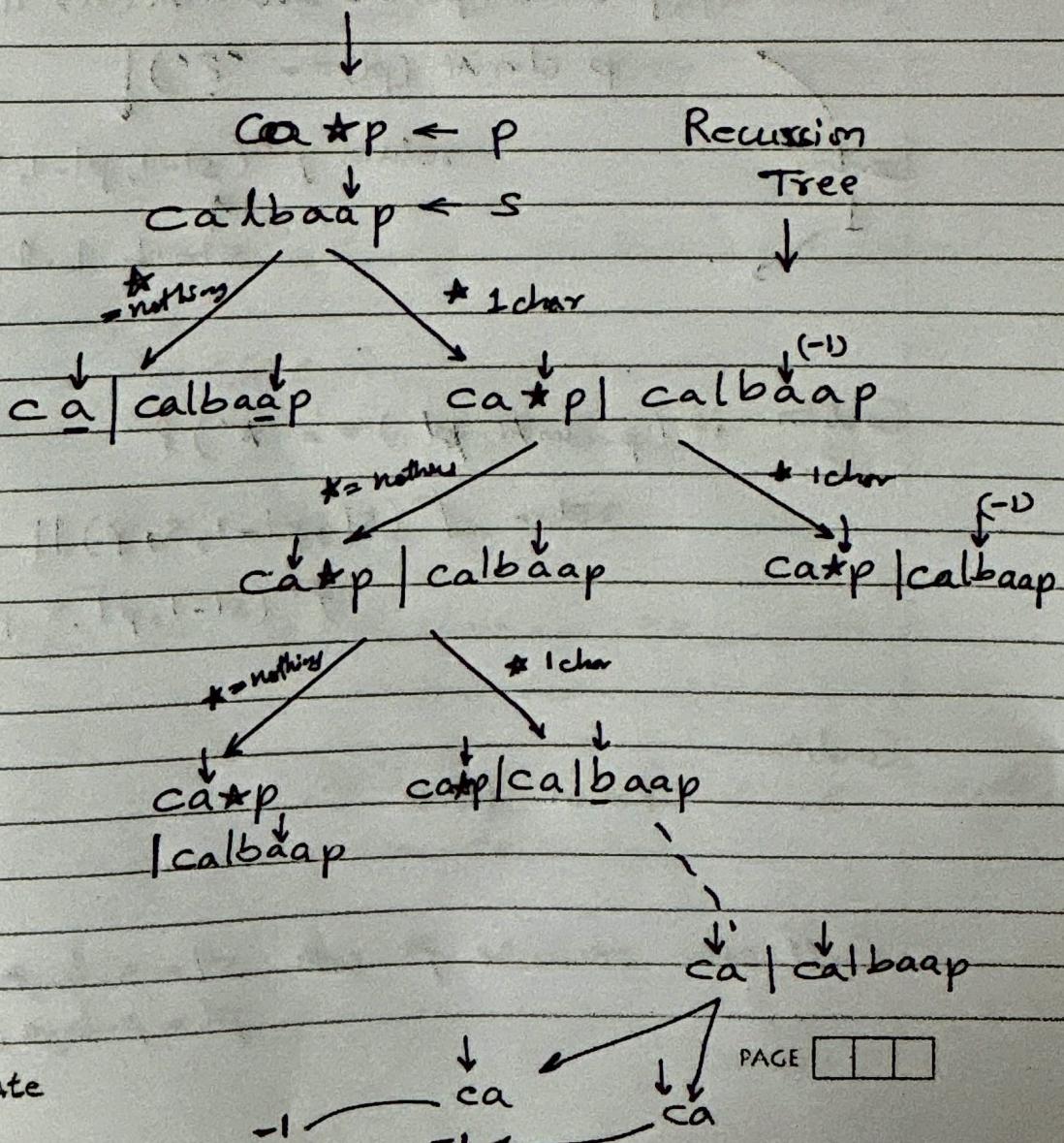
p = "aplaba"

## Approach 1. Recursion

Case 1) Match char  $\rightarrow$  using pointers keeping at last indices of S & P

Case 2) If  $p.\text{check}(\text{index}) = "?" \rightarrow$  we can try reducing / shifting indexes

(Case 3) p-Chart (index) = "★"



Recursion  $f^n$  is below

boolean  $f^n$  (int sl, int pl, String s, String p) {

// base case

if ( $sl < 0 \& pl < 0$ )  $\leftarrow$  both exhausted  
return true;

if ( $pl < 0 \& sl \geq 0$ )  $\leftarrow$  pattern only exhausted  
return false;

Base  
Condition

if ( $sl < 0 \& pl \geq 0$ )  $\leftarrow$

$\downarrow$  check if all remaining pl characters  
are == \* else return false

if ( $p.\text{charAt}(pl) == s.\text{charAt}(sl)$  ||

$p.\text{charAt}(pl) == "?"$ )

return  $f^n(sl-1, pl-1, s, p)$ ;

$\uparrow$   
shrink both sl & pl  
indexes

Condition 1. if ( $p.\text{charAt}(pl) == "*"$ ) {

2. return  $f^n(sl, pl-1, s, p)$  ||

$f^n(sl-1, pl, s, p)$ ;

Condition

3. Return false

Call above recursive fn with  $sl = s.length() - 1$  &  
 $pl = p.length() - 1$ ;

## Approach 2 Memoization

Create `int[][] dp = new int[s1][p1];`

fill all elements as -1

Call recursive f" as

$$\text{int matched} = \text{fn}(sl-1, pl-1, s, p, dp);$$

1

`fn` return type is now changed to int instead of boolean.

int f<sup>n</sup> (int sl, int pl, Story s, Story p) {

- 1)  $\hookrightarrow$  Base case is same  $\rightarrow$  True  $\rightarrow$  return 1  
False  $\rightarrow$  return 0

- 2) Check if overlapping problems are solved already as

if(dp[s][p1] != -1)  
    return dp[s][p1];

- 3) Change the conditions as →

Condition 1 → if ( $p.\text{charAt}(pl) == s.\text{charAt}(sl)$   
    ||  $p.\text{charAt}(pl) == '?'$ ) {

$dpt[s][p1] = f(sl-1, pl-1, s, p, dp);$   
 return  $dpt[s][p1];$

Condition 2 →

Condition 2 :-

if ( $p.\text{chart}(pl) == \star$ ) {

$dpt[s1][pl] = (f(s1-1, pl, s, p, dp) == 1$

$) || f(s1, pl-1, s, p, dp) == 1)?$

$1 : 0;$

}

Condition 3:  $dpt[s1][pl] == 0$

Return  $dpt[s1][pl]$  finally.