



RECURSION Playlist...

Leetcode
- 779 ✓
~~Medium~~ ✓
Easy ✓

Video -

1



K-th Symbol in

Grammar

Facebook }
Instagram } → code story with MIK

(Twitter) → CS with MIK

code story with MIK →



Company :- Google

779. K-th Symbol in Grammar

Medium

2.8K

312

Companies

We build a table of n rows (**1-indexed**). We start by writing 0 in the 1^{st} row. Now in every subsequent row, we look at the previous row and replace each occurrence of 0 with 01 , and each occurrence of 1 with 10 .

- For example, for $n = 3$, the 1^{st} row is 0 , the 2^{nd} row is 01 , and the 3^{rd} row is 0110 .

Given two integer n and k , return the k^{th} (**1-indexed**) symbol in the n^{th} row of a table of n rows.

Example 1:

Input: $n = 1, k = 1$

Output: 0

Explanation: row 1: 0

Example 2:

Input: $n = 2, k = 1$

Output: 0

Explanation:

row 1: 0

row 2: 01

Example 3:

Input: $n = 2, k = 2$

Output: 1

Explanation:

row 1: 0

row 2: 01

Brute Force

$$n = 3, \underline{k = 3}$$

(n-1)

0	0		
1	0	1	
2	0	1	1 0

2-d vector.
TLE
MLE

Recursive Approach

$n=1 \rightarrow \underline{0}$ $K=1$

$\leftarrow 1 = 2^{n-1}$

$n=2 \rightarrow \underline{0} \quad \underline{1}$ $K=1 \quad K=2$

$\leftarrow 2 = 2^{n-1}$

$n=3 \rightarrow \underline{0} \quad \underline{1} \quad \underline{1} \quad \underline{0}$ $K=1 \quad K=2 \quad K=3 \quad K=4$

Flip $\leftarrow 2^{n-1}$

$n=3$
 $k=3$

$5 - mid = 5 - 4 = 1$
 $6 - mid = 6 - 4 = 2$

$n=4$
 2^3

$\rightarrow \underline{0} \quad \underline{1} \quad \underline{1} \quad \underline{0} \quad \underline{1} \quad \underline{0} \quad \underline{0} \quad \underline{1}$ $K=1 \quad K=2 \quad K=3 \quad K=4 \quad K=5 \quad K=6 \quad K=7 \quad K=8$

$\text{length} = 8/2 = 4$

$n=4, \quad n-1, \text{flip}$

Recursion Template :-

$\begin{matrix} \backslash & \backslash \\ \cancel{x} & , & \cancel{x} \end{matrix}$
→ Solve (n, K) {

if (n == 1 && K == 1) ✓
return 0;

n = 4, K = 6

n, n-1, n-2, ... 1

T.C :-
 $O(n)$

S.C =
 $O(n)$

length = pow(2, n-1); // 2
mid = length/2; // 1

if (K <= mid) {

return Solve($\frac{n-1}{2}$, $\frac{K}{2}$);

} else {

return ⁰ ! Solve(n-1, K-mid);
}

}

Recursion Flow :-

