4th symbol Grammerwe have given input n &KGrammerdesired

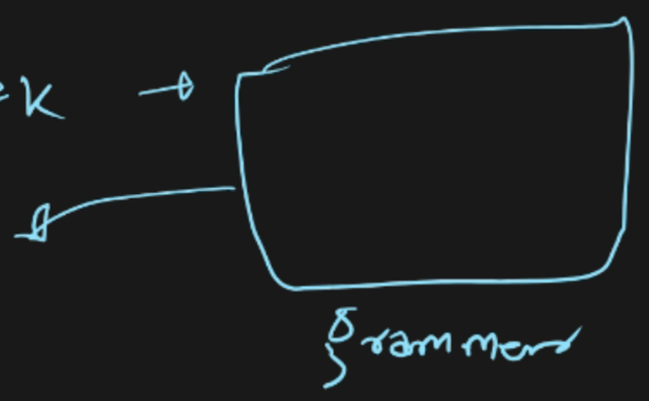
Kth Symbol Grammer

→ Problem statement → IP. OP

→ (IBH) Induction → Base Condition → Hypothesis

→ Code

Problem statement :- we have given input n & k

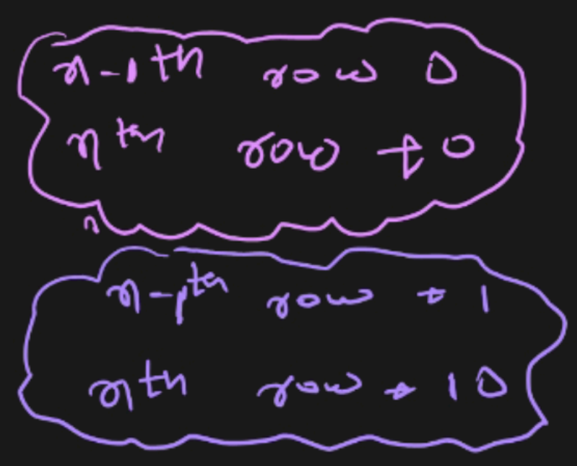


n = 1 , k = 1

return 0

defined
n → row
k → col

0th → 0
1th → 10



Ex:- n = 1 , k = 1

Output = 0

Ex:- n = 2 , k = 1

n = 1 0

n = 2 (0) 1

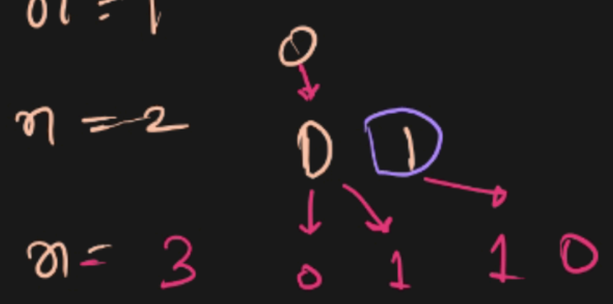
output = 0

Ex:- n = 2 , k = 2
k = 1

n = 1

n = 2

n = 3



output = 1

we have to return kth symbol in nth row
(1-index)

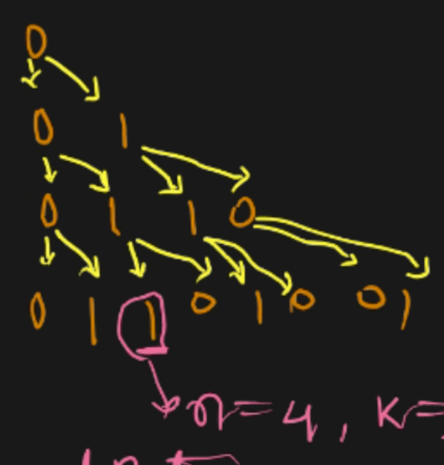
Ex: n = 4 , k = 3

n = 1

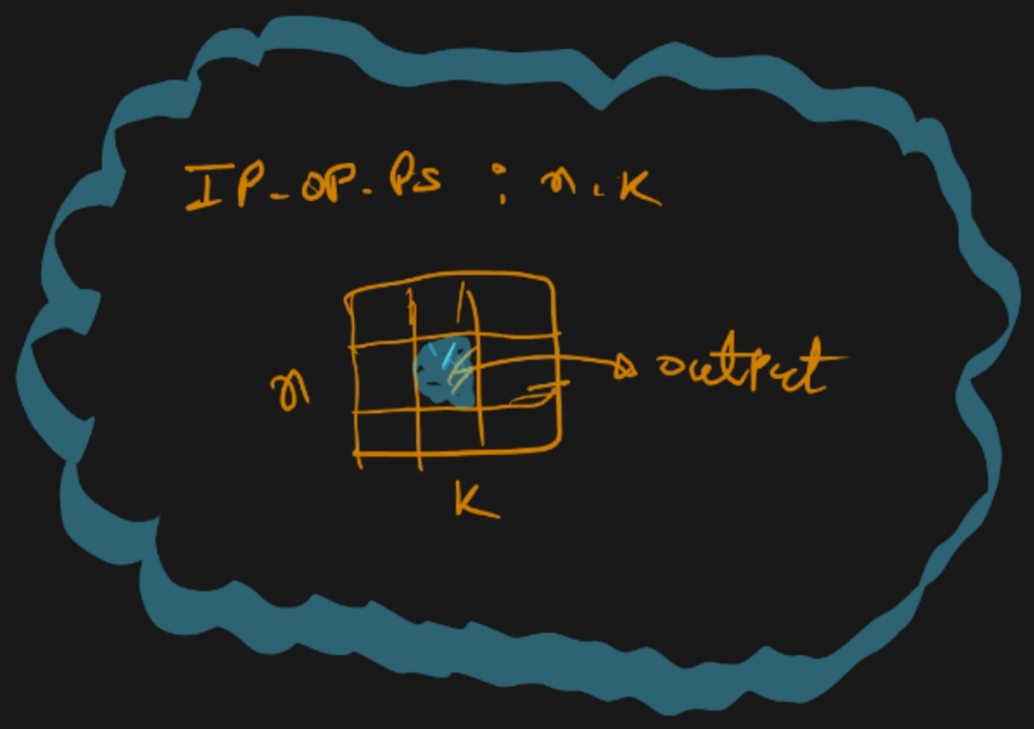
n = 2

n = 3

n = 4



output = 1



IBH Method :-

Hypothesis N, K

Solve(N, K) ⟶ 0 or 1

Small(I)

↳ Solve (N-1, ?)

Gramm

n = 1 0 → 1

n = 2 0 1 → 2

n = 3 0 1 10 → Compliment → 21

n-1

n = 4

0 1 1 0 | 1 0 0 1 → 8

Observation

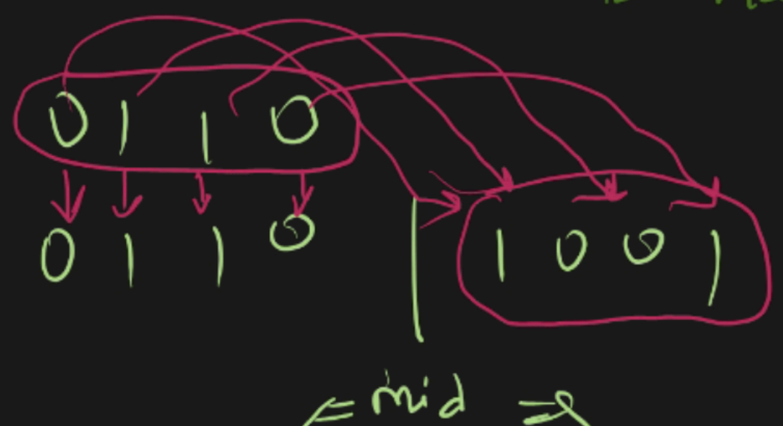
(a) length 2 × time

(b) nth row → 1st half k = n-1 row k

(c) nth row → 2nd half k = compliment of n-1 row k value

n = 3

n = 4



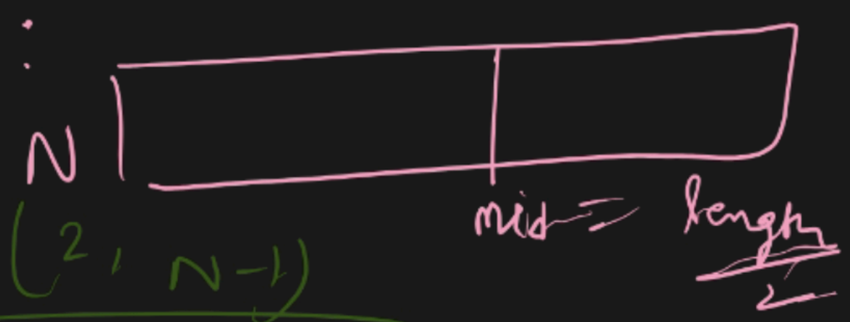

```
if(k <= mid)
    Solve(N,k) → Solve(N-1,k)
if(k > mid)
    Solve(N,k) → Solve(N-1, k-mid)
```

→ Hypothesis

Base condition :- n = 1 , k = 1

return 0;

Induction Step :



int mid = Pow(2, N-1) / 2

if (k <= mid) return solve(N-1, k);

else return ! solve(N-1, k - mid);

N=1 D → 1 = 2^0

N=2 D 1 → 2 = 2^1

N=3 D 1 1 0 → 4 = 2^2

N=4 D 1 1 0 1 0 0 1 → 8 = 2^3

if N = 1 , length = 2^{N-1}