4

	r	B	g
h_1	1	5	7
h_2	5	8	4
h_3	3	2	9
h_4	1	2	4

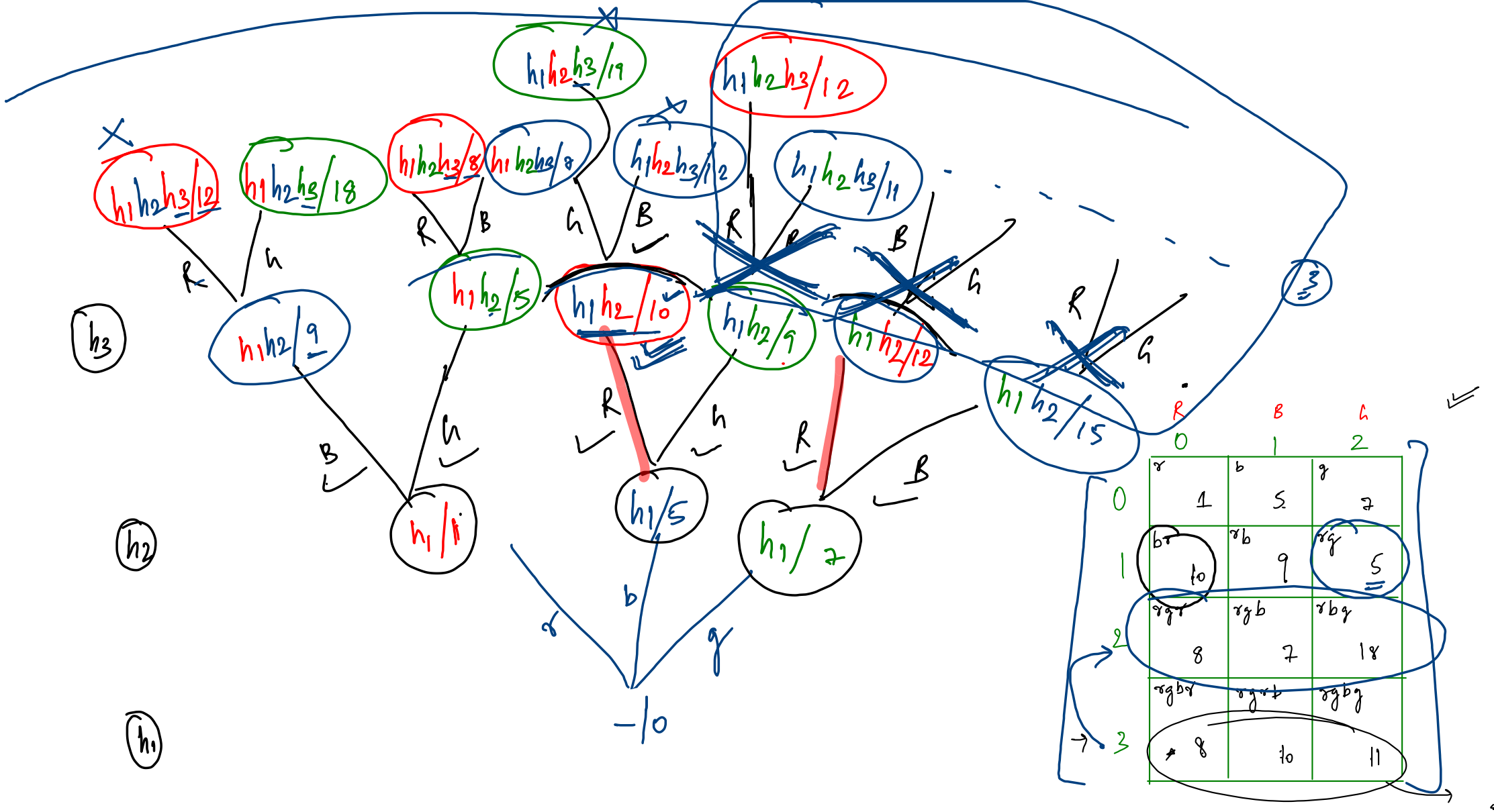
H_1	H_2	H_3	H_4
R	B	R	G
G	B	R	B

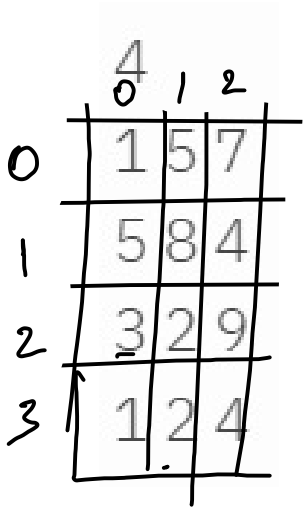
3 colours

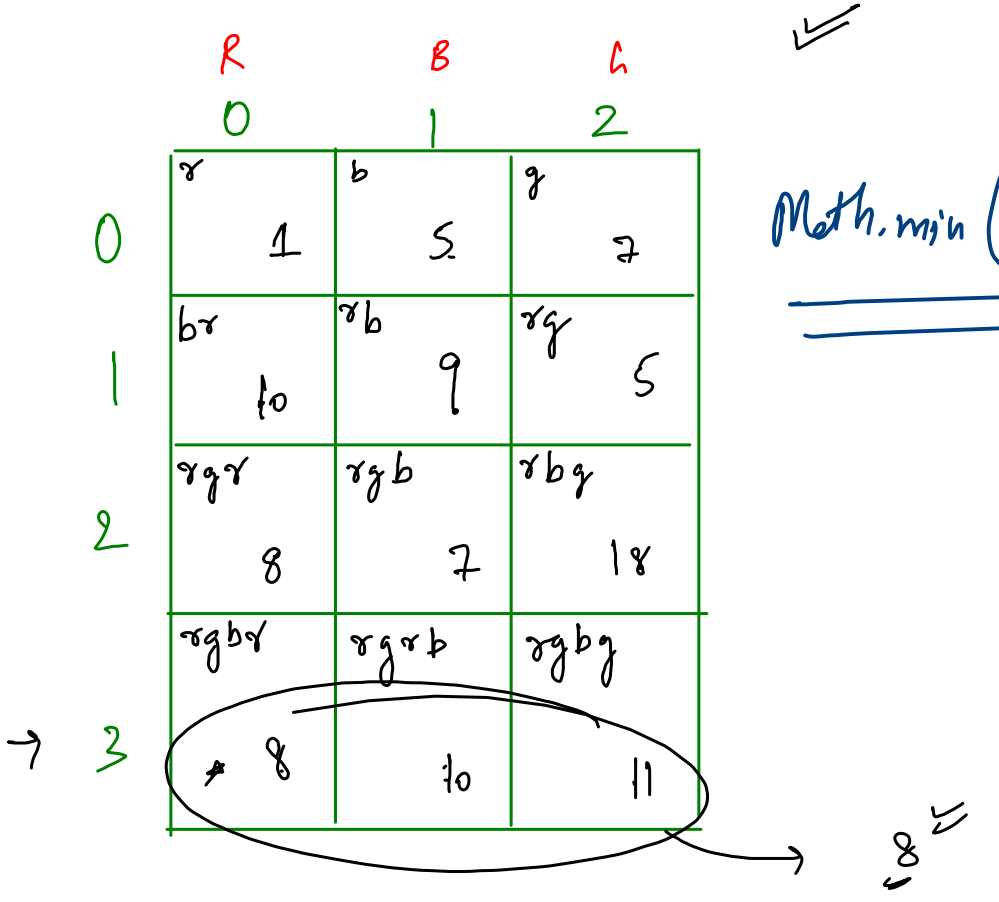
r b g

Minimum amount ?

↳ (paint all houses provided no consecutive/ adjacent house can have same color)


h1h2h3/19
h1h2h3/12
h1h2h3/12
h1h2h3/18
h1h2h3/8
h1h2h3/8
h1h2h3/12
h1h2h3/11
h3
h1h2/9
h1h2/5
h1h2/10
h1h2/9
h1h2/12
h1h2/15
h2
h1/8
h1/5
h1/7
h1
-10
R
G
B
r
b
g
0
1
2
3
8
10
11





Math.min(Math.min(a, b), c)

8

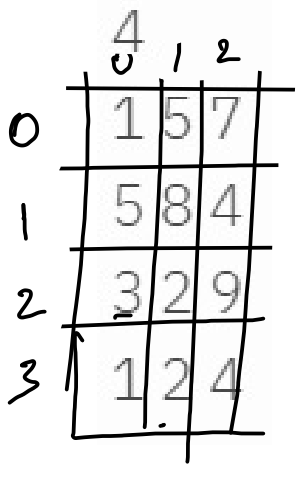
```java
// logic
int dp[][] = new int[n][3];

for(int h = 0 ; h < n ; h++){
    if(h == 0){
        dp[h][0] = cost[h][0];
        dp[h][1] = cost[h][1];
        dp[h][2] = cost[h][2];
    }else{
        dp[h][0] = Math.min(dp[h-1][1],dp[h-1][2]) + cost[h][0];
        dp[h][1] = Math.min(dp[h-1][0],dp[h-1][2]) + cost[h][1];
        dp[h][2] = Math.min(dp[h-1][1],dp[h-1][0]) + cost[h][2];
    }
}

int minCost = Math.min( Math.min(dp[n-1][0],dp[n-1][1]) , dp[n-1][2] );
System.out.println(minCost);
```

T.C. => O(n)

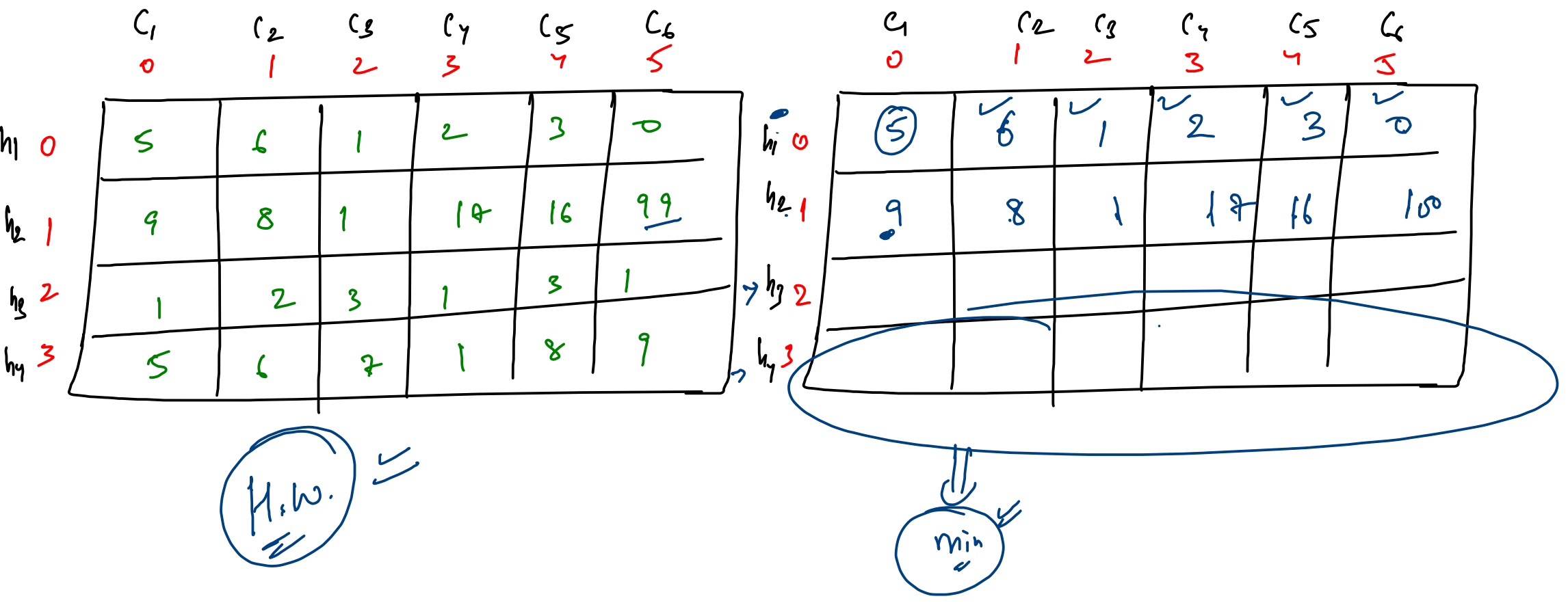
S.C. => O(n)

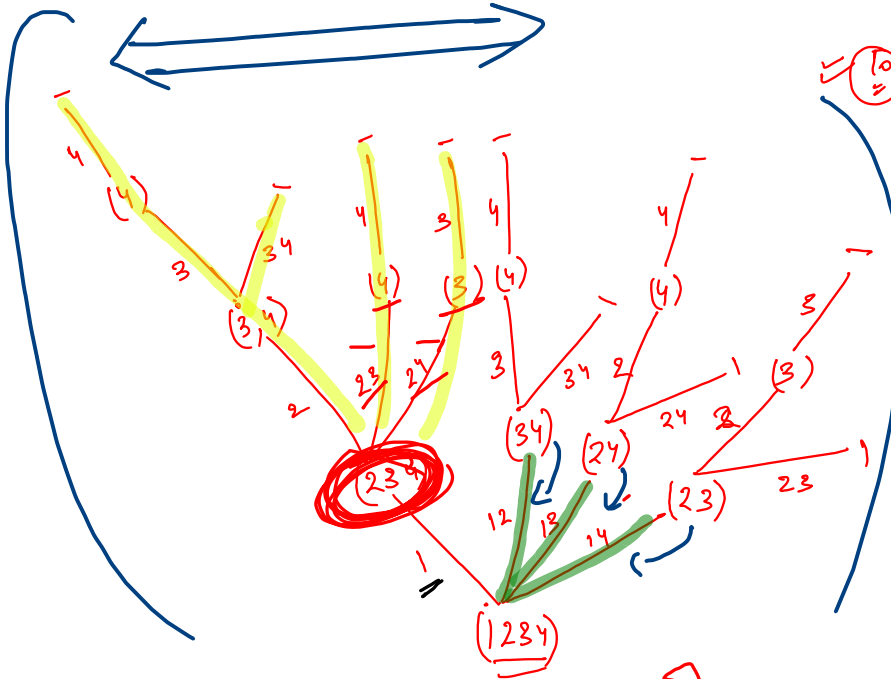


Cost

dp =

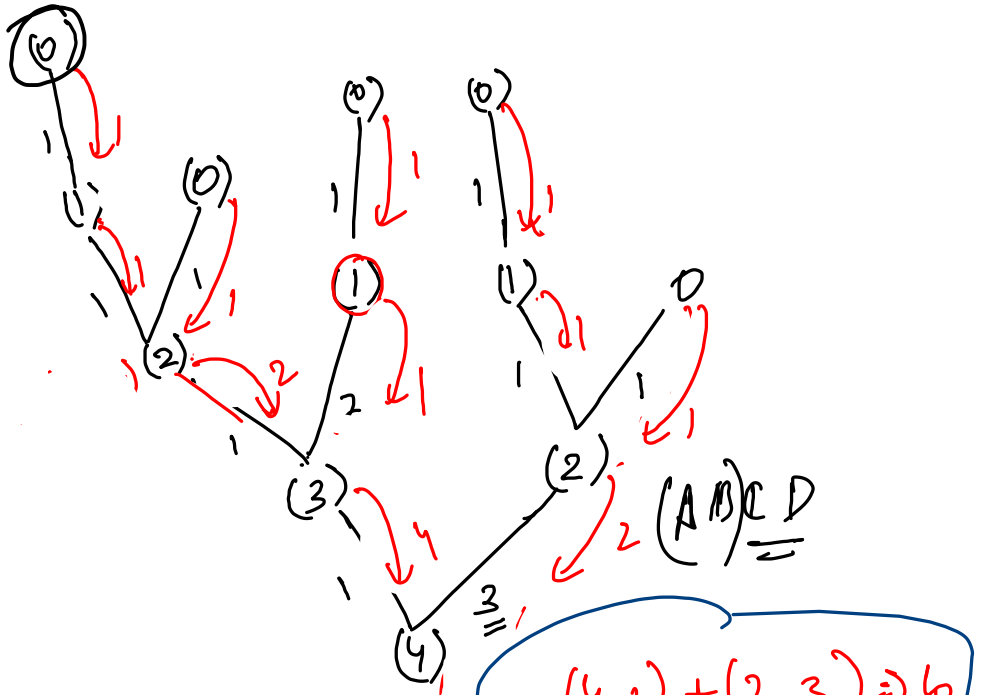
	0	1	2
0	1	5	7
1	10	9	5
2	8	7	18
3	—	—	—


C1 C2 C3 C4 C5 C6
0 1 2 3 4 5
h1 0 5 6 1 2 3 0
h2 1 9 8 1 10 16 99
h3 2 1 2 3 1 3 1
h4 3 5 6 7 1 8 9
H.W.
C1 C2 C3 C4 C5 C6
0 1 2 3 4 5
h1 0 5 6 1 2 3 0
h2 1 9 8 1 17 16 100
h3 2
h4 3
min



10 =
1-2-3-4
1-2-34
1-23-4
1-24-3
12-3-4
12-34
13-2-4
13-24
14-2-3
14-23

234 →
2-3-4
2-34
23-4
24-3



(4,1) + (2,3) ⇒ 6

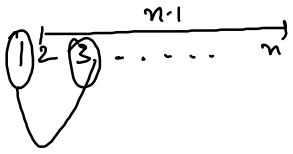
1 Pen → 10 Rs

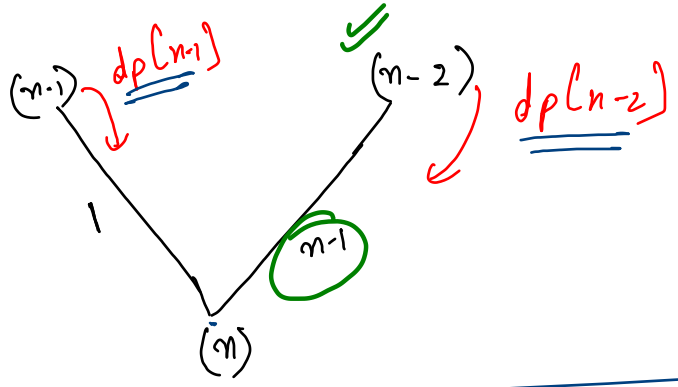
2 × 1 Pen → 20 Rs

1 Pen . 1 Pen 1 Pen → 30

3 Pen → 30

Fibonacci

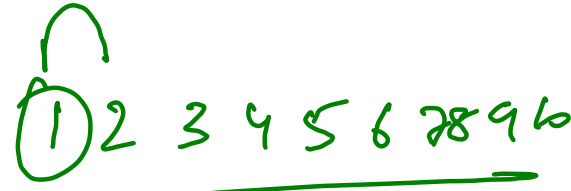




$$dp[n] = dp[n-1] + [dp[n-2] \cdot (n-1)]$$

n=4

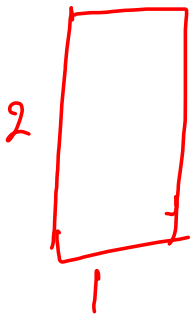
0	1	2	3	4
1	1	2	4	10

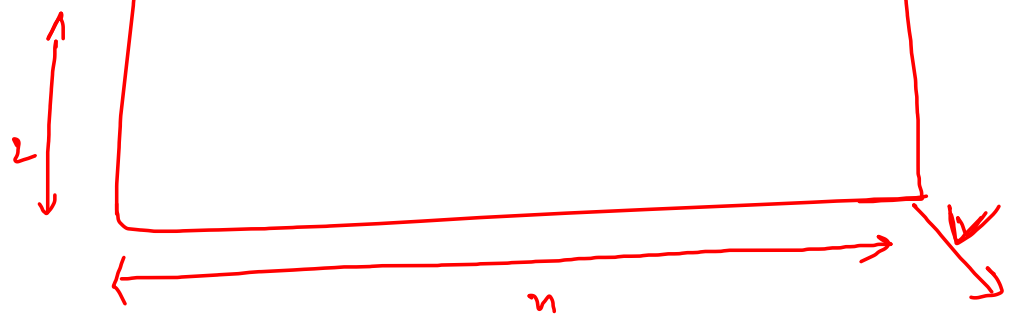


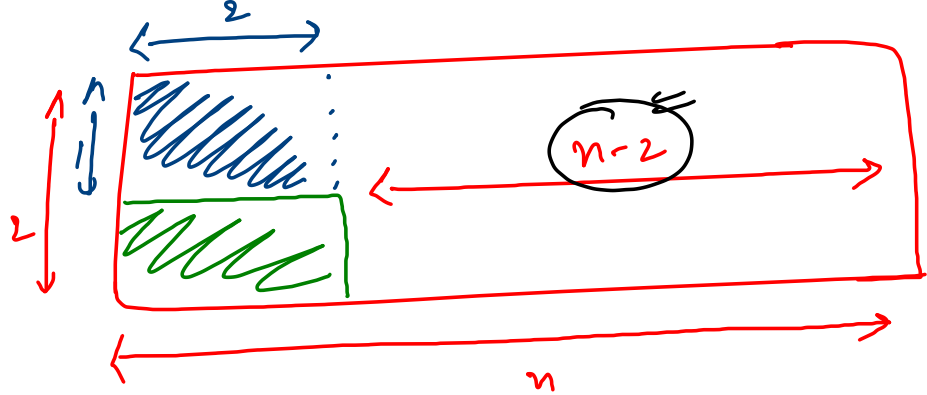
9

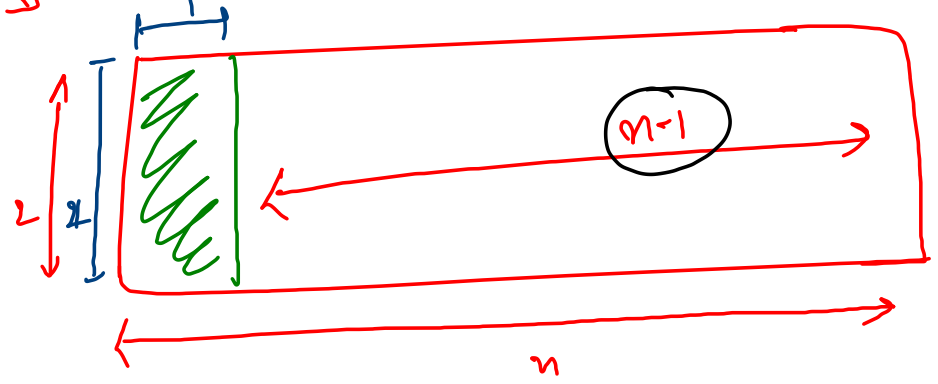
8

$n = 8$



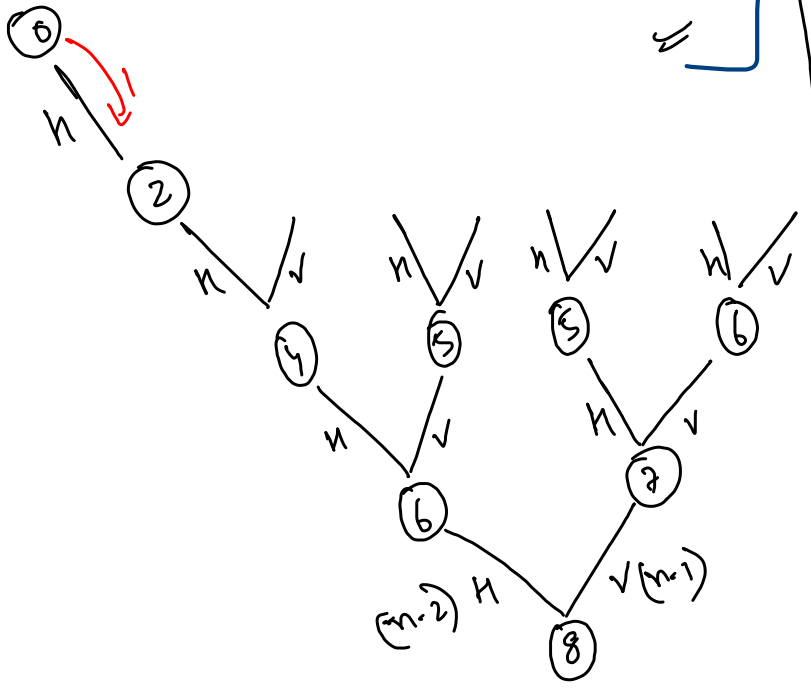






0	1	2	3	4	5	6	7	8
1	1	2	3	5	8	13	21	34

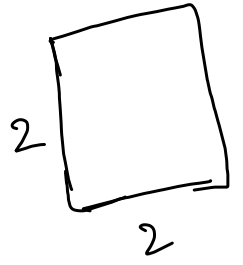
H.W.

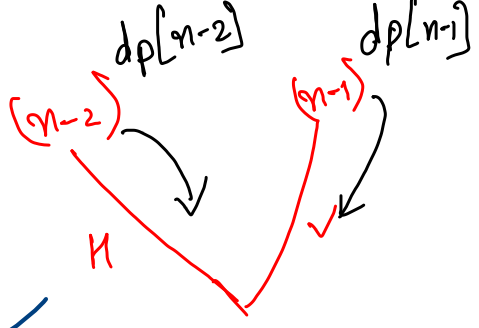


n=0, 1 (kuch nahi karna)

n=1, 1 (vertically)

n=2,

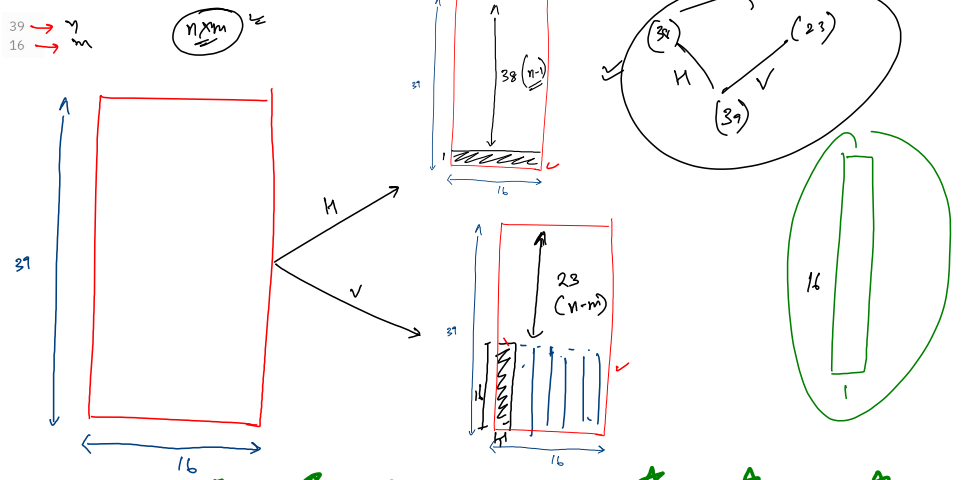


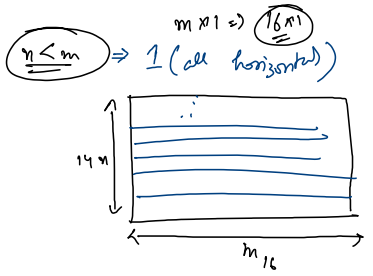


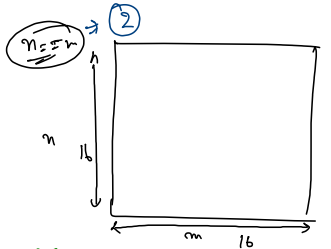
$$dp[n] = dp[n-2] + dp[n-1]$$

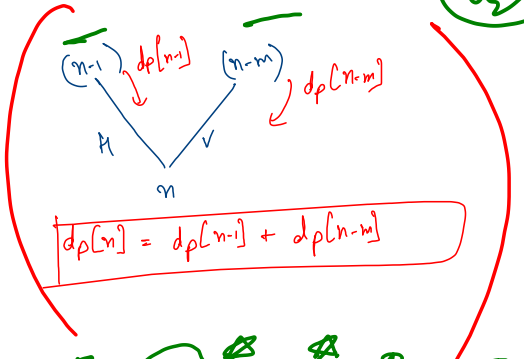
m <= 16

dp(n,m)









dp

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	2	3	4	5	6

21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39
7	8	9	10	11	12	13	14	15	16	17	19	22	26	31	37	44	52	61

i : 0 → n

i < m
↳ dp[i] = 1

i == m
↳ dp[i] = 2

i > m
↳ dp[i] = dp[i-1] + dp[i-m]


Paint House - Many Colors (M, N)
Tiling With 2 * 1 Tiles
Tiling With M * 1 Tiles (M, N)
discussed
Friends Pairing
Paint House
Easy 10 Auth 0 Public Sol 17
Easy 10 Auth 0 Public Sol 18
Easy 10 Auth 0 Public Sol 19
Easy 10 Auth 0 Public Sol 20
Easy 10 Auth 0 Public Sol 21