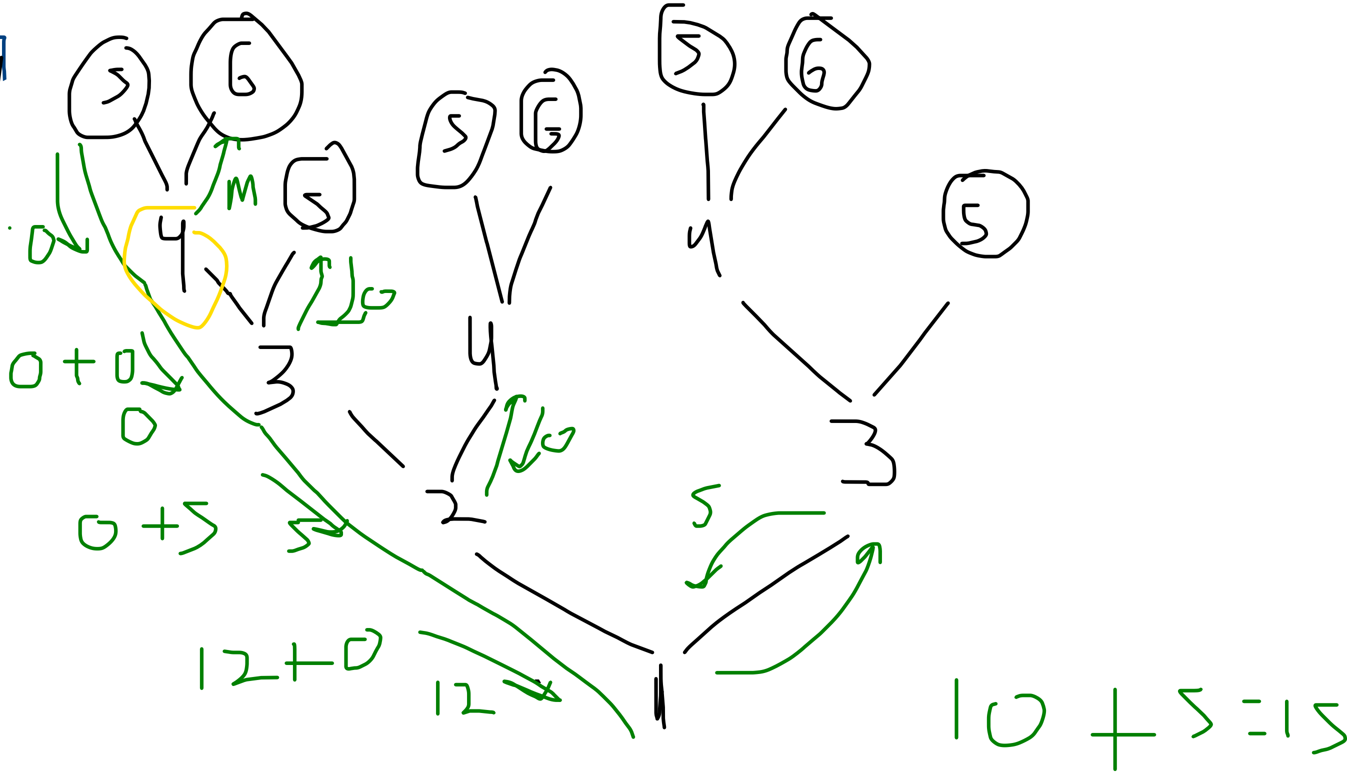
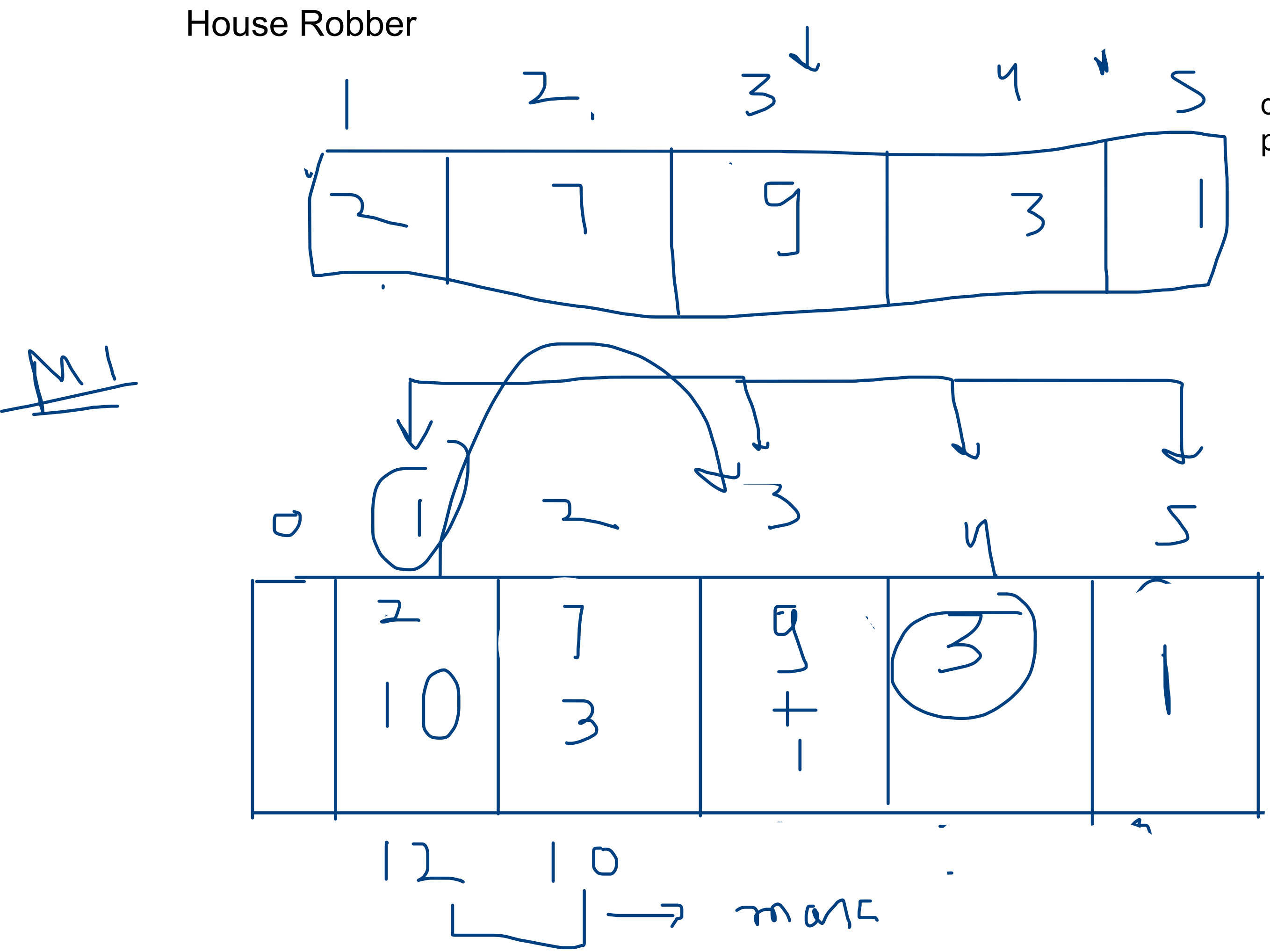
Min Cost stair path -->
Memoization

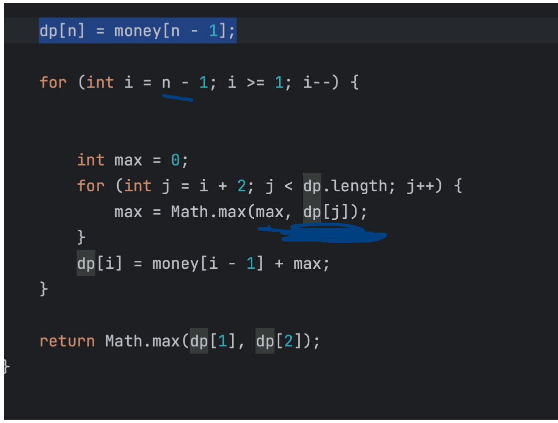
0	1	2	3	4
10	12	5	0	10
1	2	3	4	5



House Robber



dp[i]--> if we rob ith house then max profit we can make till 5th house

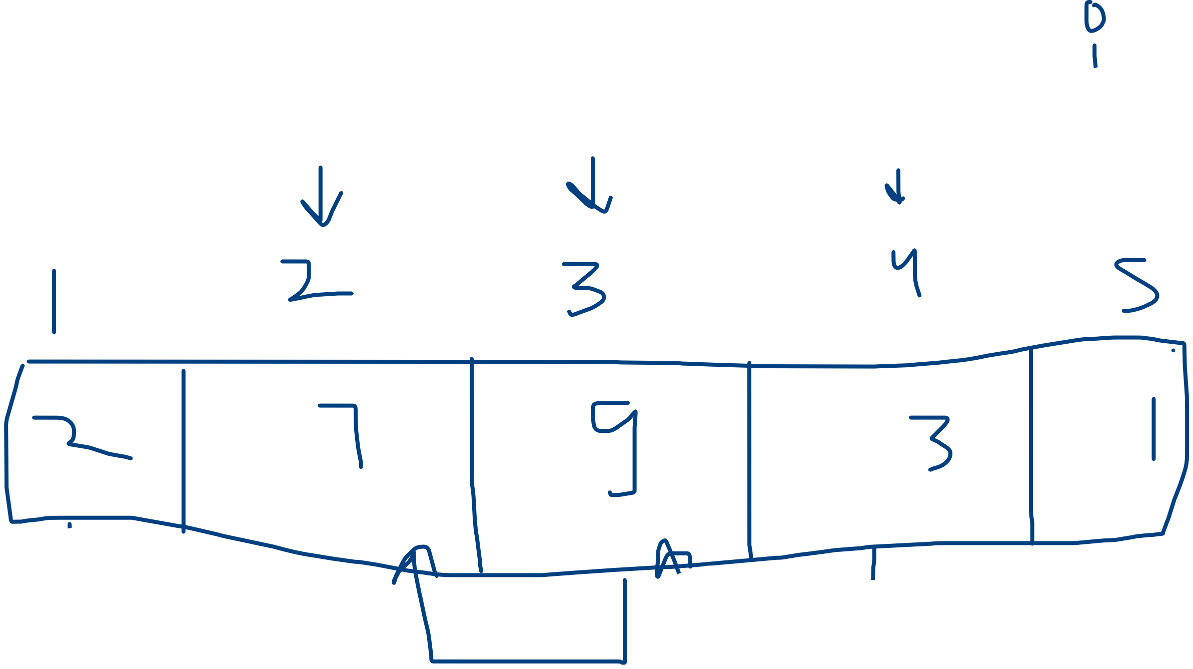


```
dp[n] = money[n - 1];

for (int i = n - 1; i >= 1; i--) {


    int max = 0;
    for (int j = i + 2; j < dp.length; j++) {
        max = Math.max(max, dp[j]);
    }
    dp[i] = money[i - 1] + max;
}

return Math.max(dp[1], dp[2]);
```

include

exclude

problems

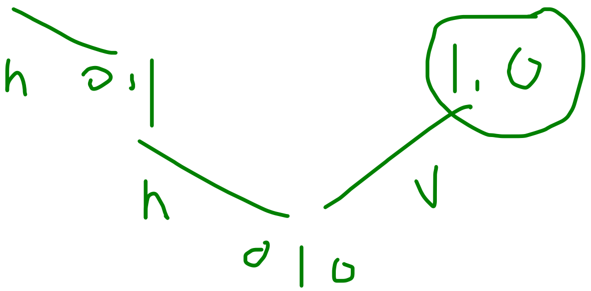


include	2✓	7+0	9+2	3+7
exclude	0✓	0+(2,0)	0+(7,2)	0+(11,7)

max(include, exclude)

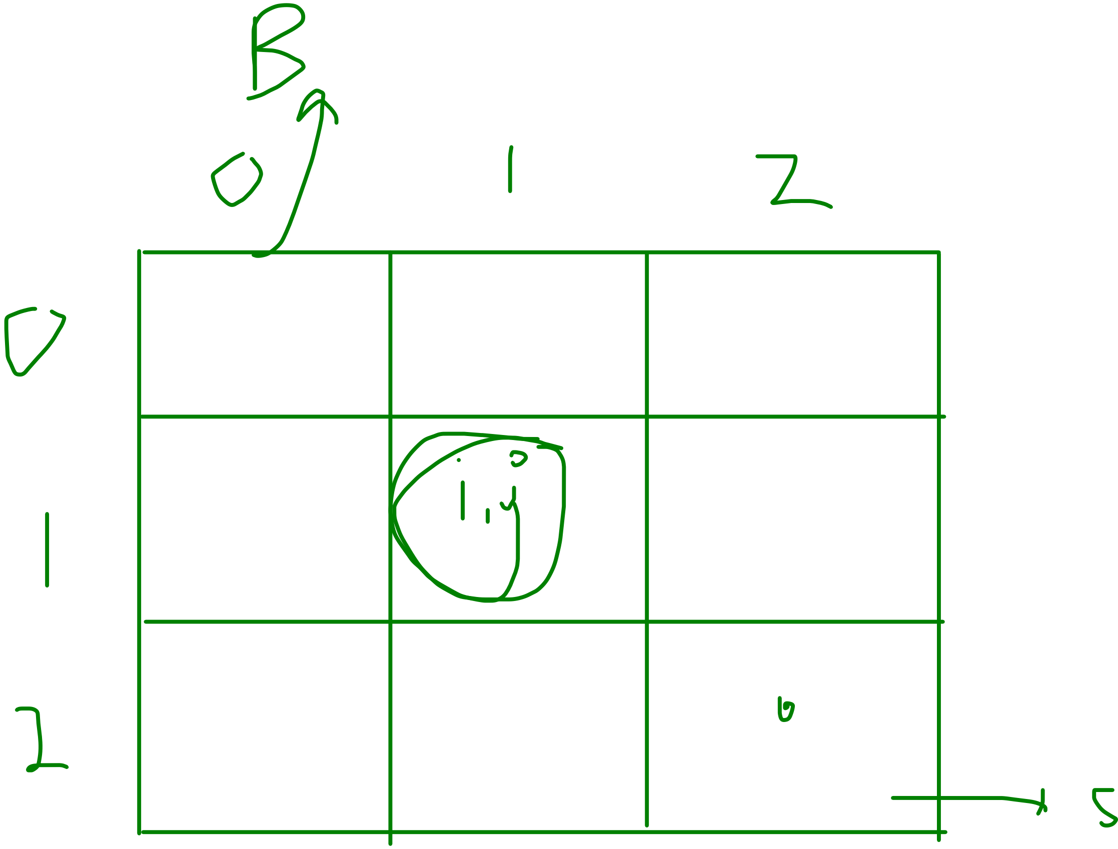
Minimum Path Sum

	0	1	2
0	1	3	1
1	1	5	1
2	4	2	1



2D

$$\begin{array}{c|ccc} & 0 & 1 & 2 \\ \hline 0 & 1 & 3 & 1 \\ 1 & 1 & 5 & 1 \\ 2 & 4 & 2 & 1 \end{array}$$



dp[i][j] --> min cost need to reach destination (m, n) from (i, j)

	0	1	2
0	1	3	1
1	1	5	1
2	4	2	1

B

	0	1	2
0	1+5 6	3+2	1+1
1	1+(6,4) 7	5+(1,4) 5+1	1+0
2	4+2	2+0	

7 7 x

0,0 → 2,2