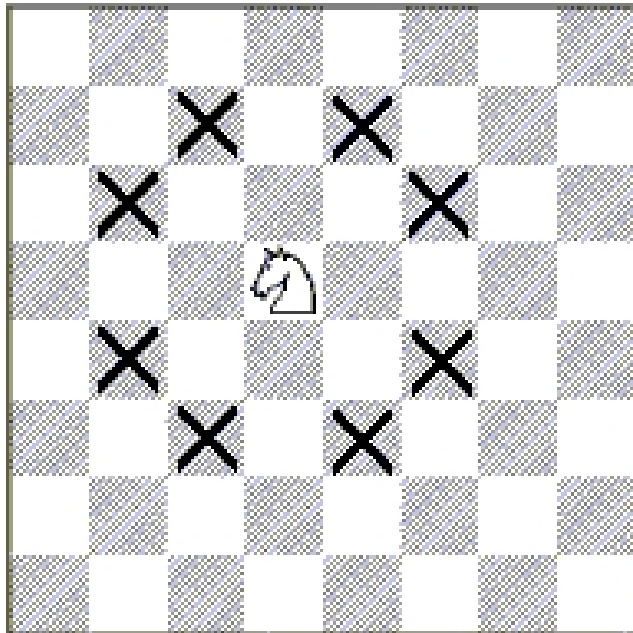


# Recursion Instances

## Knight Tourist (EASY)

- 走馬步（騎士走訪）
- 如何能夠在  $N * N$  的西洋棋棋盤的最上角放置一個西洋棋的馬，請輸出一種可能的順序，使得它可以在不重複走某個格子的情況下走完全部的格子。

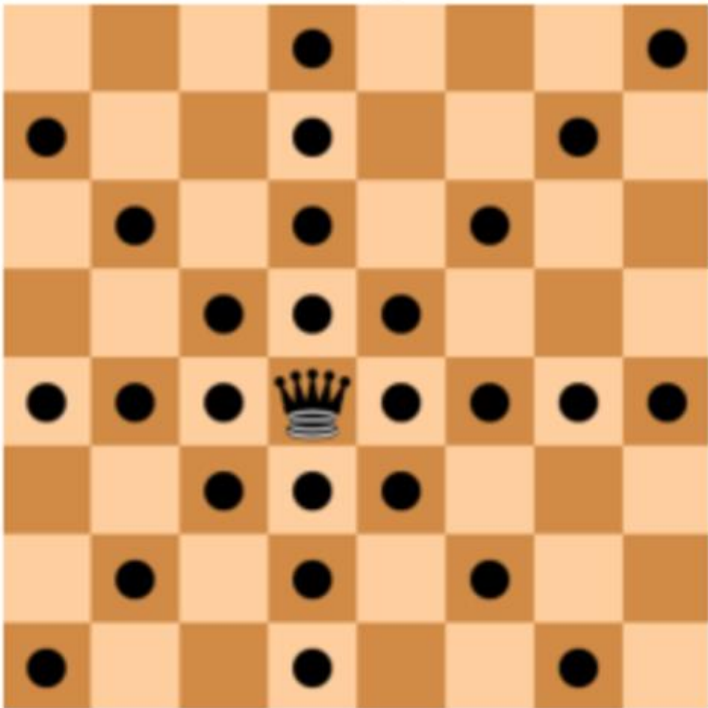


- 簡單版本：  $N \leq 6$ （正常實作）
- 進階版本：  $N \leq 16$ （ Warnsdorff's Rule 剪枝手法）

# Recursion Instances

## Chessboard and Queens (EASY)

- 如何能夠在  $8 * 8$  的西洋棋棋盤放置兩個西洋棋的皇后，請輸出能所有它們不可互相攻擊的擺法數。
- <https://cses.fi/problemset/task/1624>



# Recursion Instances

## Apple Division (EASY)

- 有一堆不同重量的蘋果，數量不超過 20，請分成兩堆使得兩堆總重量相同。
- <https://cses.fi/problemset/task/1623>

# Recursion Instances

## Gray Code (EASY ~ MEDIUM)

- Gray Code 是任意兩個相鄰數的代碼只有一位二進制數不同的編碼。
- 請輸入一個  $n$ ，輸出從 1 到  $2^n$  的一種可能的 Gray Code。
- 注意：有很多種合法的 Gray Code。
- $n \leq 16$
- <https://cses.fi/problemset/task/2205>

十進位    格雷碼    二進位

|   |     |     |
|---|-----|-----|
| 0 | 000 | 000 |
| 1 | 001 | 001 |
| 2 | 011 | 010 |
| 3 | 010 | 011 |
| 4 | 110 | 100 |
| 5 | 111 | 101 |
| 6 | 101 | 110 |
| 7 | 100 | 111 |

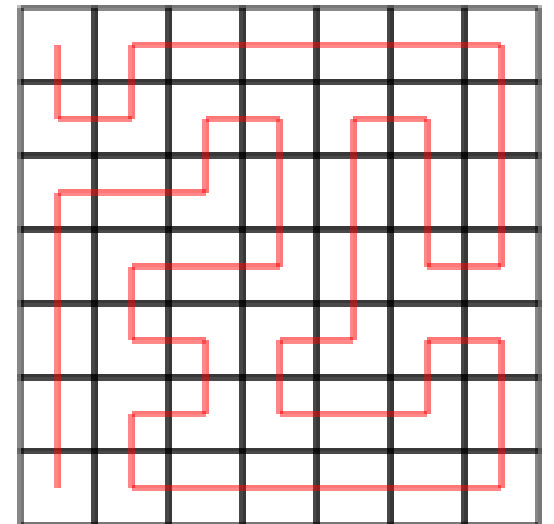
# Recursion Instances

## Grid Paths (EASY ~ MEDIUM)

- 給定一個  $7 * 7$  的棋盤，從左上走到右下，每次可以往上下左右四個方向走，但你被限定在某些步驟只能往特定方向，請輸出有多少種可能的走法。

- ?????R????U????????????????????LD????D?

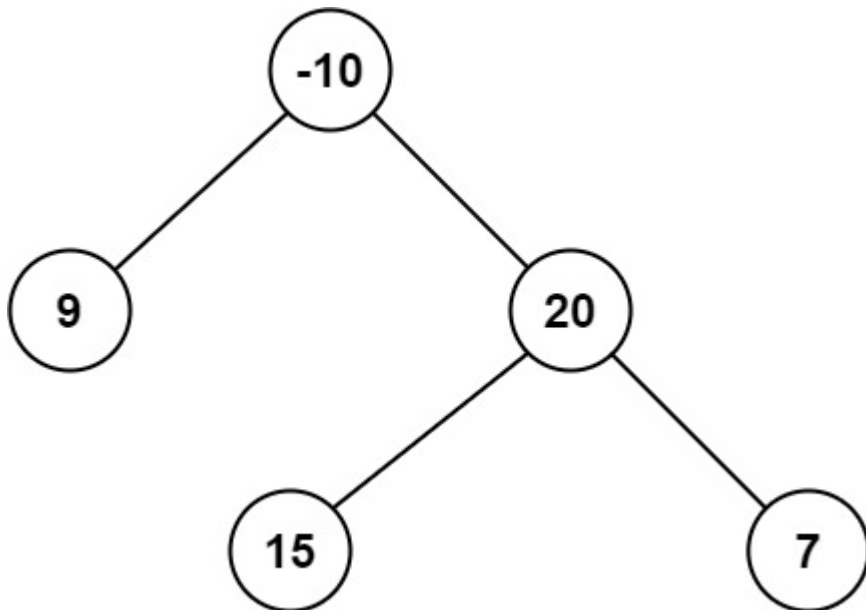
- <https://cses.fi/problemset/task/1625/>



# Recursion Instances

## Regex Matching (MEDIUM~HARD)

- 給一個二元樹，找一條總和最大的路徑。
- 例如下圖的答案為  $15 + 20 + 7 = 42$ 。
- <https://leetcode.com/problems/binary-tree-maximum-path-sum/>



### Constraints:

- The number of nodes in the tree is in the range  $[1, 3 * 10^4]$ .
- $-1000 \leq \text{Node.val} \leq 1000$

# Recursion Instances

## Regex Matching (MEDIUM+Memorize)

- 給一個  $M * N$  矩陣，求一個最長的嚴格遞增的路徑長度。
- <https://leetcode.com/problems/longest-increasing-path-in-a-matrix/>

|   |   |   |
|---|---|---|
| 9 | 9 | 4 |
| 6 | 6 | 8 |
| 2 | 1 | 1 |

Diagram illustrating a 3x3 matrix with values and arrows indicating a path:

- Row 1: 9, 9, 4
- Row 2: 6, 6, 8
- Row 3: 2, 1, 1

Arrows indicate a path starting from the bottom-left cell (2) and moving up to the top-left cell (9):

- From (3,1) to (2,1): up arrow
- From (2,1) to (1,1): up arrow

### Constraints:

- `m == matrix.length`
- `n == matrix[i].length`
- `1 <= m, n <= 200`
- `0 <= matrix[i][j] <= 231 - 1`

# Recursion Instances

## Regex Matching (MEDIUM~HARD)

- ‘.’ 可以匹配任意單個字元，‘\*’ 可以匹配前面一個元素 0 次以上。
- 給你一個字串  $s$  和一個 Regex Pattern  $p$ ，問  $p$  是否可以表達  $s$ 。
- <https://leetcode.com/problems/regular-expression-matching/>

### Example 2:

**Input:**  $s = "aa"$ ,  $p = "a^*"$

**Output:** true

**Explanation:** ‘\*’ means zero or more of the preceding element, ‘a’. Therefore, by repeating ‘a’ once, it becomes “aa”.

### Example 3:

**Input:**  $s = "ab"$ ,  $p = ".^*"$

**Output:** true

**Explanation:** “.\*” means “zero or more (\*) of any character (.)”.

### Constraints:

- $1 \leq s.length \leq 20$
- $1 \leq p.length \leq 20$



# Recursion Instances

## Regex Matching (HARD+Trie) (WON'T EXPLAIN)

- 給一個  $M * N$  矩陣，和一個裝着許多字串的 `words` 陣列，回傳那些可以在板子上找到的字（一個格子在同一個字內只能使用一次）。
- <https://leetcode.com/problems/word-search-ii>

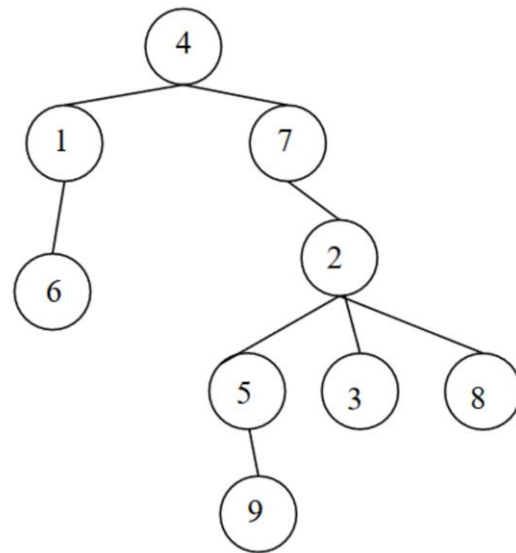
|   |   |   |   |
|---|---|---|---|
| o | a | a | n |
| e | t | a | e |
| i | h | k | r |
| i | f | l | v |

### Constraints:

- `m == board.length`
- `n == board[i].length`
- `1 <= m, n <= 12`
- `board[i][j]` is a lowercase English letter.
- `1 <= words.length <= 3 * 104`
- `1 <= words[i].length <= 10`
- `words[i]` consists of lowercase English letters.
- All the strings of `words` are unique.

# Recursion Instances

## Tree Analyses (EASY) (WON'T EXPLAIN)



樹狀圖中的兩個節點  $u$  和  $v$  之間的距離  $d(u,v)$  定義為兩節點之間邊的數量。如圖一中， $d(7, 5) = 2$ ，而  $d(1, 2) = 3$ 。對於樹狀圖中的節點  $v$ ，我們以  $h(v)$  代表節點  $v$  的高度，其定義是節點  $v$  和節點  $v$  下面最遠的葉節點之間的距離，而葉節點的高度定義為 0。如圖一中，節點 6 的高度為 0，節點 2 的高度為 2，而節點 4 的高度為 4。此外，我們定義  $H(T)$  為  $T$  中所有節點的高度總和，也就是說  $H(T) = \sum_{v \in T} h(v)$ 。給定一個樹狀圖  $T$ ，請找出  $T$  的根節點以及高度總和  $H(T)$ 。

### 輸入說明

第一行有一個正整數  $n$  代表樹狀圖的節點個數，節點的編號為 1 到  $n$ 。  
接下來有  $n$  行，第  $i$  行的第一個數字  $k$  代表節點  $i$  有  $k$  個子節點，第  $i$  行接下來的  $k$  個數字就是這些子節點的編號。  
每一行的相鄰數字間以空白隔開。

### 範例輸入 #1

範例一：  
7  
0  
2 6 7  
2 1 4  
0  
2 3 2  
0  
0

- <https://zerojudge.tw/ShowProblem?problemid=c463>