

Week 7

Topic: Breadth-first search, BFS

322. Coin Change

- Difficulty: Medium
- Problem URL: <https://leetcode.com/problems/coin-change/description/>
- Description:
有一組不同面額的硬幣 `coins`，以及一個目標金額 `amount`。目標是找出最少的硬幣數量來湊出這個金額。其中每種面額的金幣可以無限使用，若無法湊出該金額，則回傳 `-1`。

Example1:

Input: `coins = [1,2,5]`, `amount = 11`

Output: 3

Explanation: $11 = 5 + 5 + 1$

Example2:

Input: `coins = [2]`, `amount = 3`

Output: -1

Example3:

Input: `coins = [1]`, `amount = 0`

Output: 0

詳細說明與約束條件請參考 *Leetcode* 網站。

752. Open the Lock

- Difficulty: Medium
- Problem URL: <https://leetcode.com/problems/open-the-lock/description/>
- Description:
有一個 4 位數字的轉盤鎖，每位數是 0~9，能向上或向下轉動，初始為 "0000"。你可以每次轉動任一位數 +1 或 -1，但不能轉到 deadends (死鎖狀態)。目標是從 "0000" 轉到指定的 target 組合，找出最少的轉動次數，若無法達成則回傳 "-1"。

Example1:

Input: deadends = ["0201","0101","0102","1212","2002"], target = "0202"

Output: 6

Explanation:

A sequence of valid moves would be "0000" => "1000" => "1100" => "1200" => "1201" => "1202" => "0202".

Note that a sequence like "0000" => "0001" => "0002" => "0102" => "0202" would be invalid, because the wheels of the lock become stuck after the display becomes the dead end "0102".

Example2:

Input: deadends = ["8888"], target = "0009"

Output: 1

Explanation:

We can turn the last wheel in reverse to move from "0000" => "0009".

Example3:

Input: deadends = ["8887","8889","8878","8898","8788","8988","7888","9888"], target = "8888"

Output: -1

Explanation:

We cannot reach the target without getting stuck.

詳細說明與約束條件請參考 *Leetcode* 網站。

815. Bus Routes

- Difficulty: Hard
- Problem URL: <https://leetcode.com/problems/bus-routes/description/>
- Description:

有一個 `routes` 陣列，表示多條公車路線，`routes[i]` 是第 i 條公車路線的所有站牌，且該公車會無限循環在這些站之間。回傳從 `source` 抵達 `target` 搭乘最少的公車數量，如果無法到達，請回傳 `-1`。

例如: `routes[0] = [1, 5, 7]` 表示第 0 號公車會依序行駛:

$1 \rightarrow 5 \rightarrow 7 \rightarrow 1 \rightarrow 5 \rightarrow 7 \rightarrow \dots$ 。

初始會從某一個站牌 `source` 出發 (初始時不在任何公車上)，目標是到達某個站牌 `target`。只能透過「搭乘公車」來作移動。

Example1:

Input: `routes = [[1,2,7],[3,6,7]]`, `source = 1`, `target = 6`

Output: 2

Explanation:

The best strategy is take the first bus to the bus stop 7, then take the second bus to the bus stop 6.

Example2:

Input: `routes = [[7,12],[4,5,15],[6],[15,19],[9,12,13]]`, `source = 15`, `target = 12`

Output: -1

詳細說明與約束條件請參考 *Leetcode* 網站。