

# 演算法與程式解題實務

Mong-Jen Kao (高孟駿)

Monday 18:30 – 21:20

這是 BFS 的基礎模板。

# BFS 廣度優先探索

- BFS 是依照「與出發點間的距離」為順序來探索圖，過程中，需要使用到一個雙向佇列 (queue)  $Q$ 。
    1. 開始時，把出發點  $v$  放進  $Q$  裡，並把  $v$  標示為已探索
    2. 每個回合，從  $Q$  的前端 (front) 取出一個點 (假設為  $u$ )，並將它展開：
      - 考慮點  $u$  的所有鄰居，  
若  $u$  有鄰居還沒有探索過，  
就把它放到  $Q$  的尾端 (back)，並標示為已探索
- 重覆步驟 2，直到  $Q$  裡沒有剩下任何點為止

這是 BFS 的基礎模板。

# BFS 廣度優先探索

```
function bfs( int v ) {  
    let Q be a double-ended queue, initially empty.  
    Add v to the tail of Q and set visited[v] to be true.  
    while Q is not empty, do  
        Remove a vertex from the front of Q.  
        Let the vertex be u.  
        for each  $w \in \text{adj}(u)$ ,  
            if w is not yet visited, then  
                Add w to the tail of Q.  
                Set visited[w] to be true.  
}
```

# BFS 廣度優先探索

這是 BFS 的基礎模板。

$v$  為 BFS 探索的起點

```
function bfs( int  $v$  ) {
```

```
    let  $Q$  be a double-ended queue, initially empty.
```

```
    Add  $v$  to the tail of  $Q$  and set  $visited[v]$  to be true.
```

```
    while  $Q$  is not empty, do
```

```
        Remove a vertex from the front of  $Q$ .  
        Let the vertex be  $u$ .
```

```
        for each  $w \in adj(u)$ ,
```

```
            if  $w$  is not yet visited, then
```

```
                Add  $w$  to the tail of  $Q$ .
```

```
                Set  $visited[w]$  to be true.
```

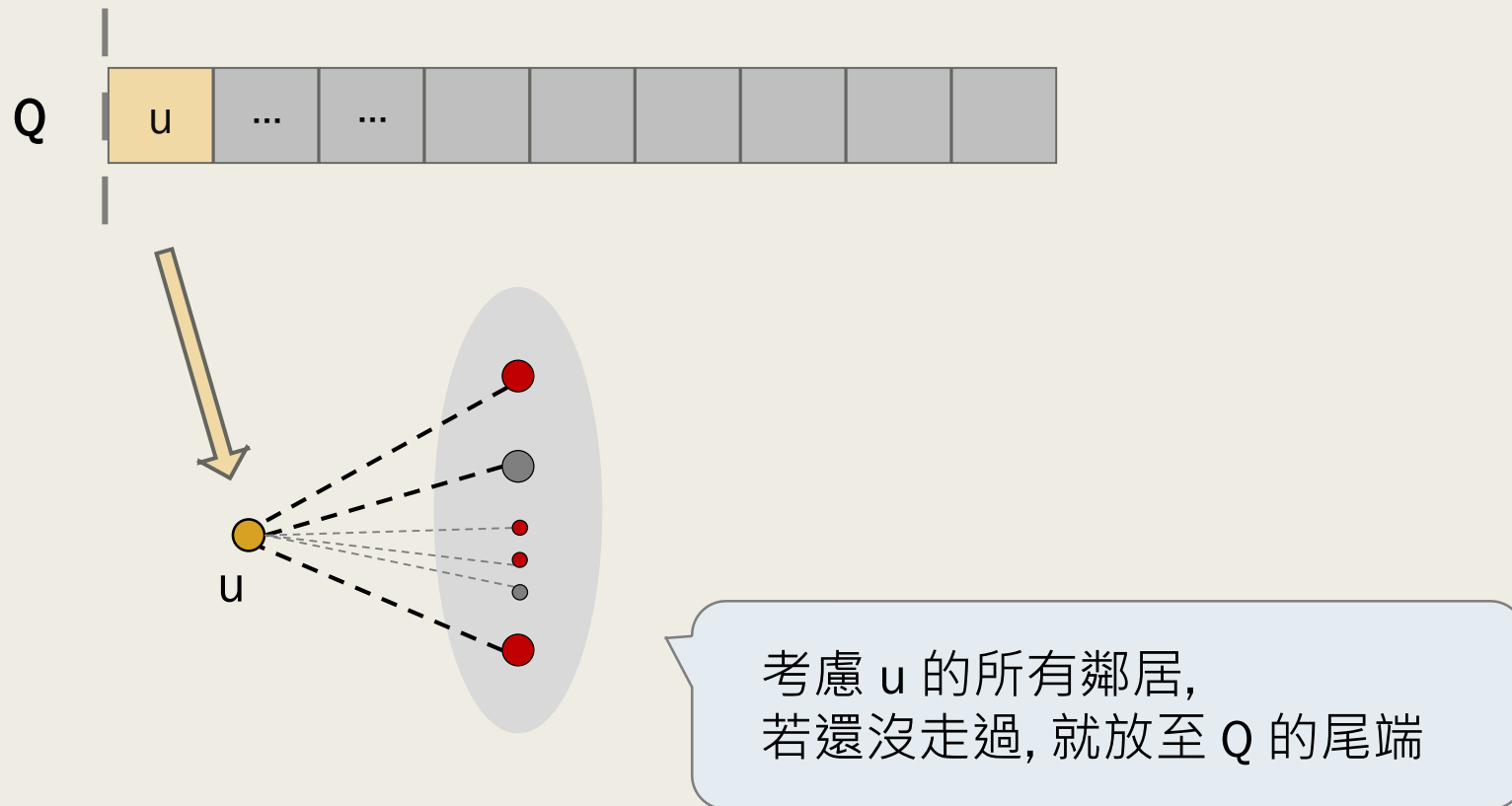
```
}
```

從佇列前端取出一個點,並將它展開.  
將還沒探索過的鄰居放到佇列尾端

在此處決定: 由  $u$  走到  $w$   
( $w$  的來源點為  $u$ )

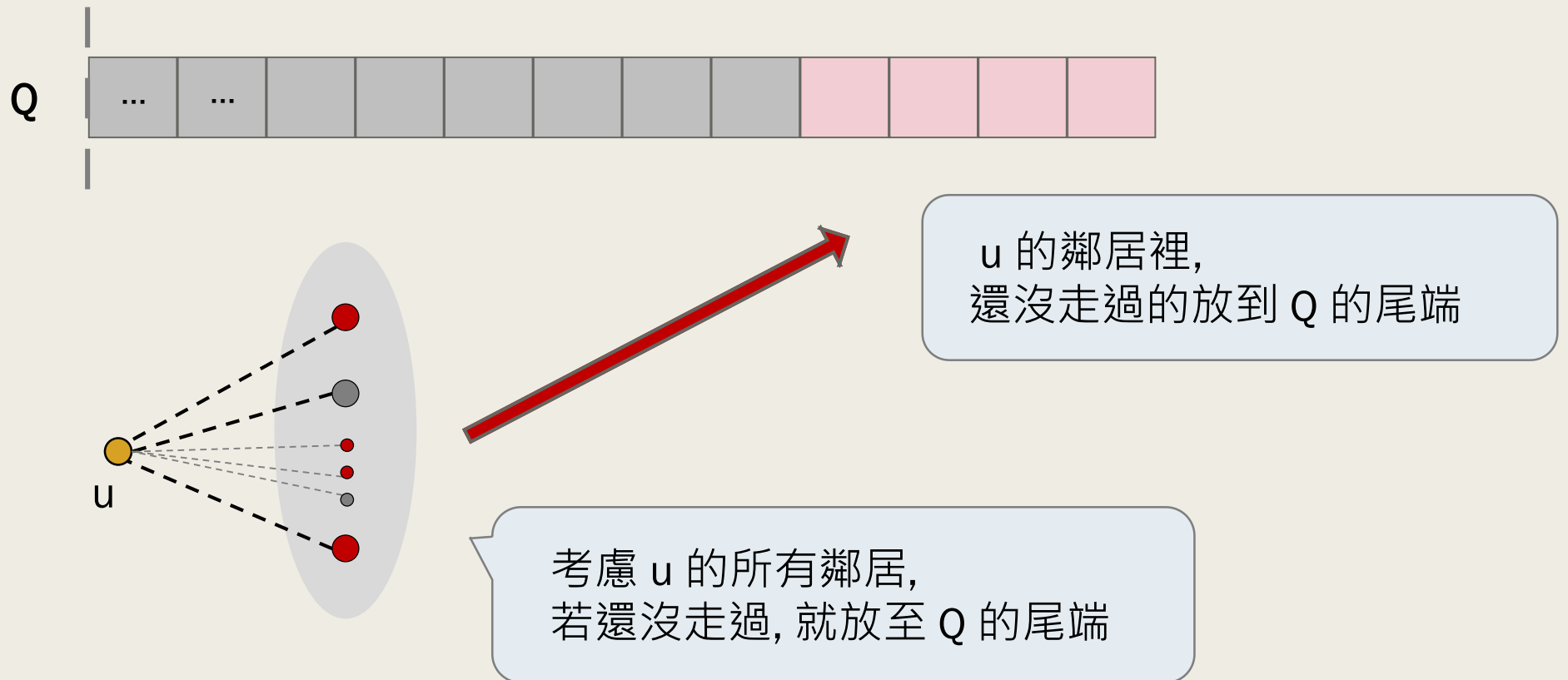
# BFS 廣度優先探索

- 每個回合, 從  $Q$  的前端 (front) 取出一個點, 並將它展開



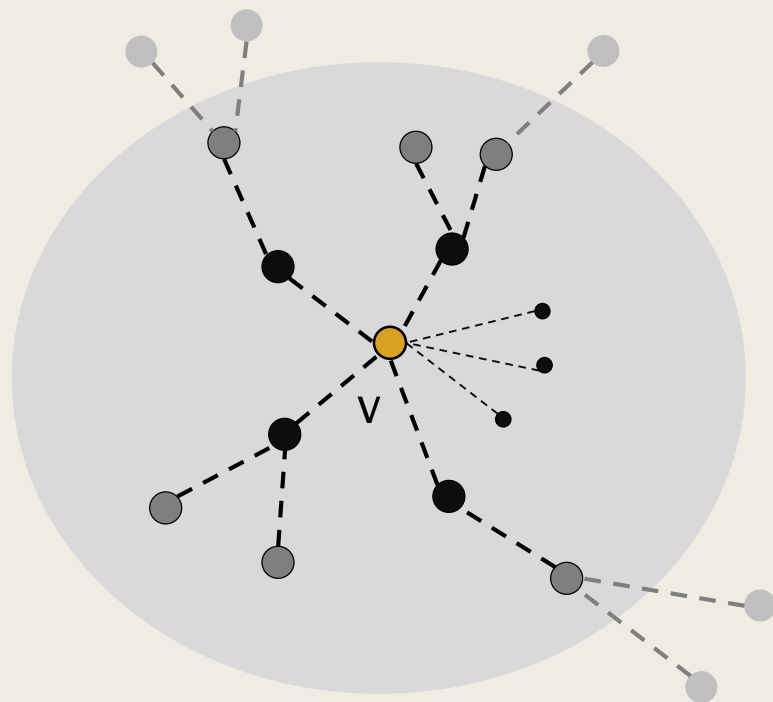
# BFS 廣度優先探索

- 每個回合, 從  $Q$  的前端 (front) 取出一個點, 並將它展開



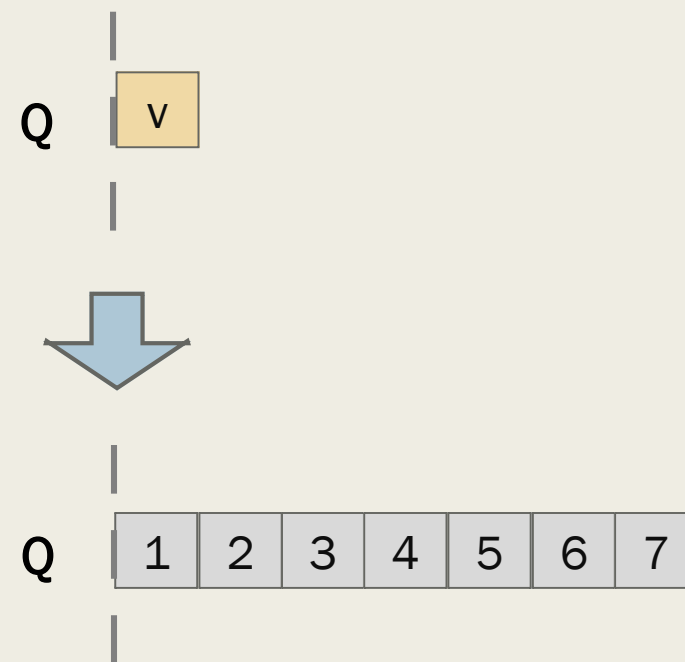
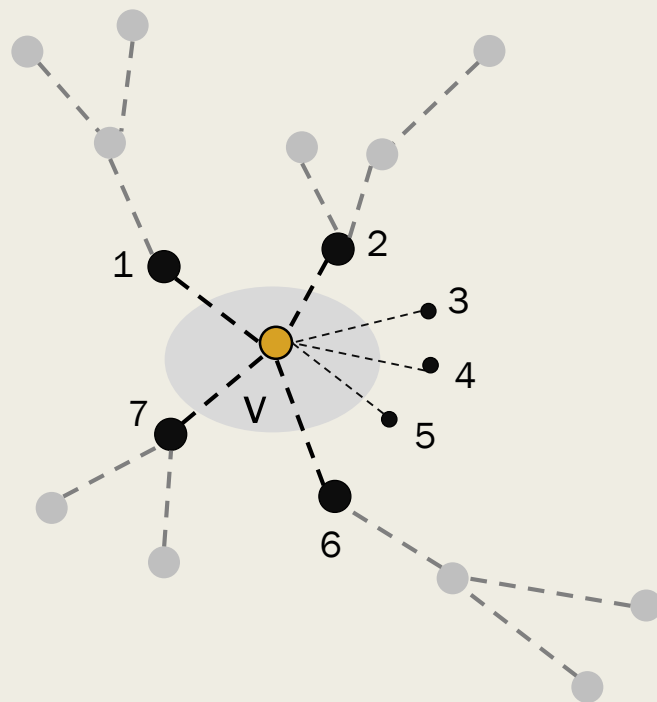
# BFS 廣度優先探索

- BFS 可以保證, 圖裡的點會依「與出發點間的距離」為順序被探索



# BFS 廣度優先探索

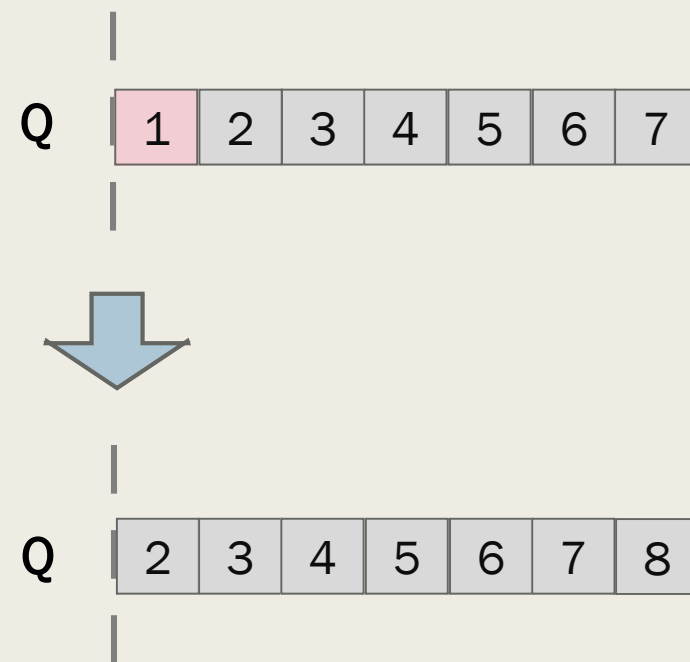
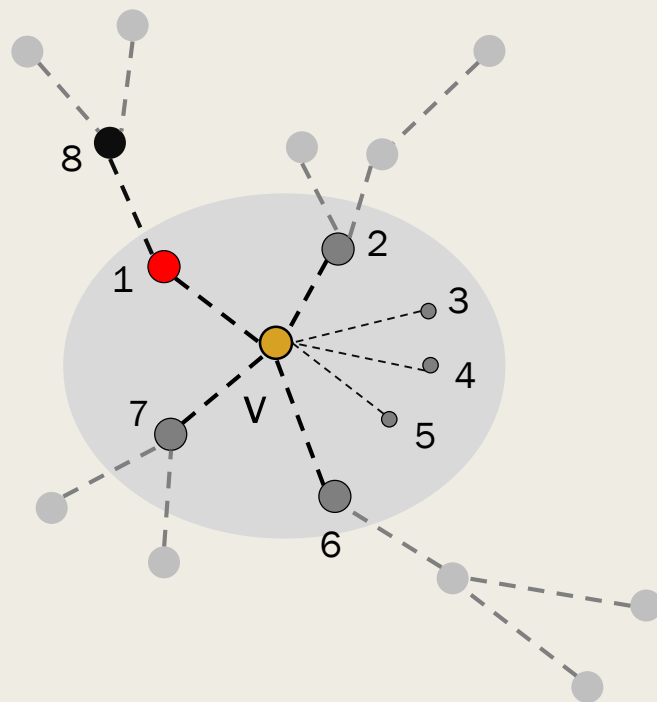
- BFS 可以保證, 圖裡的點會依「與出發點間的距離」為順序被探索





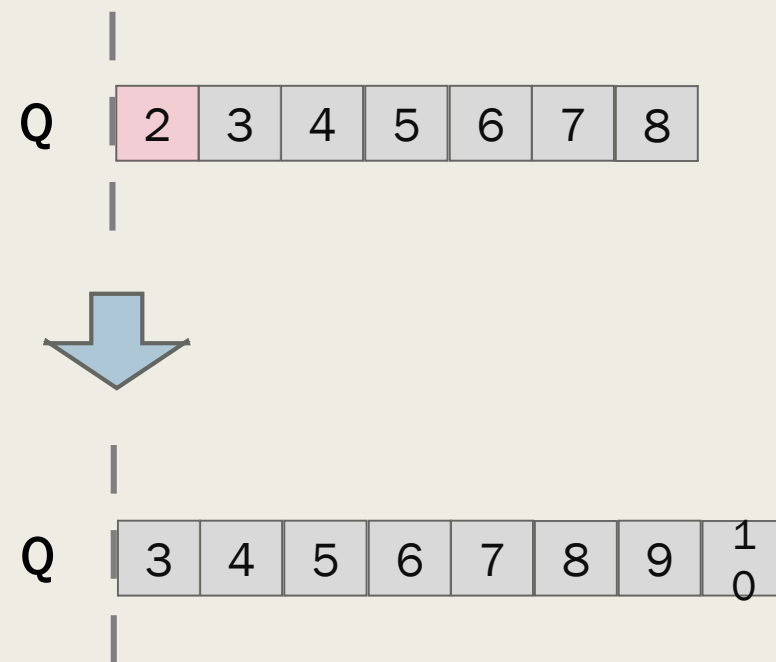
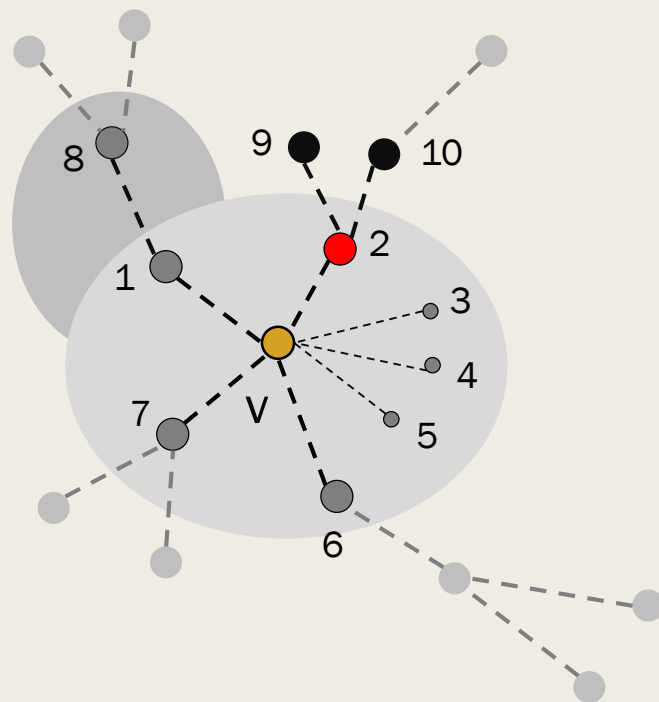
# BFS 廣度優先探索

- BFS 可以保證, 圖裡的點會依「與出發點間的距離」為順序被探索



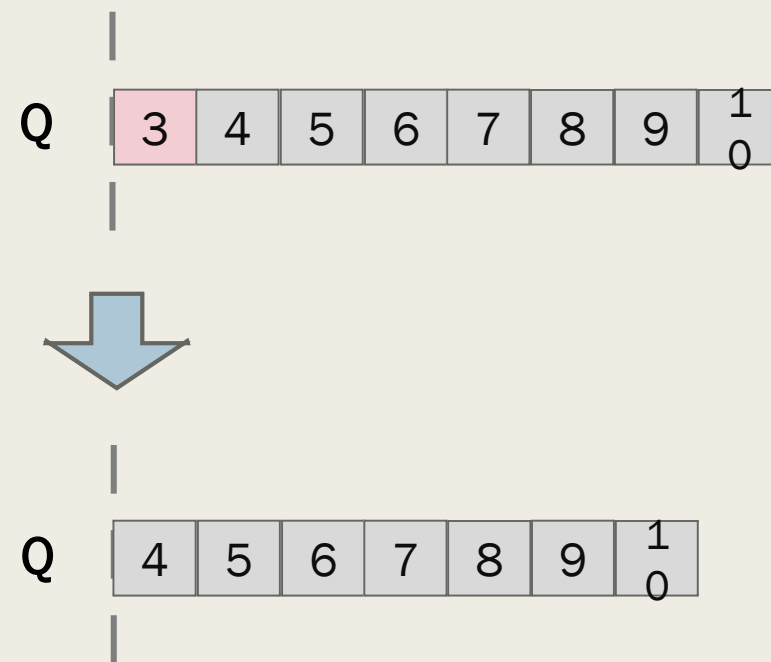
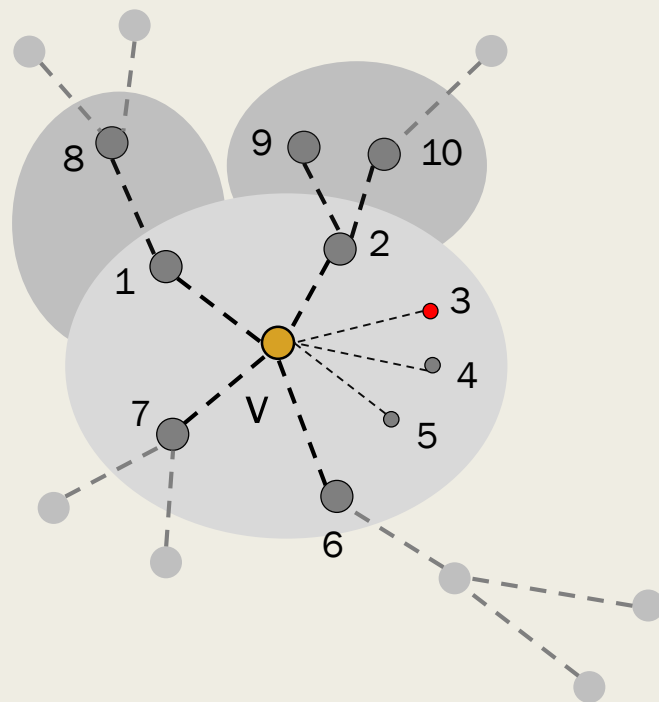
# BFS 廣度優先探索

- BFS 可以保證, 圖裡的點會依「與出發點間的距離」為順序被探索



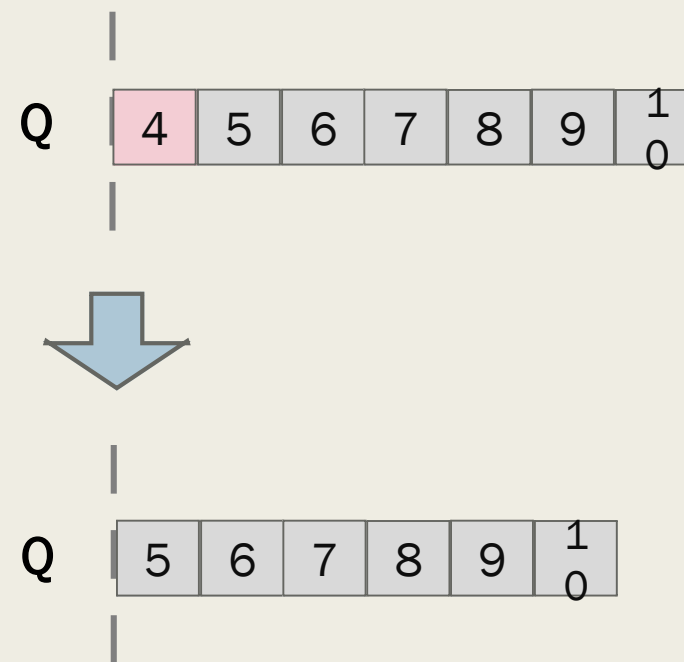
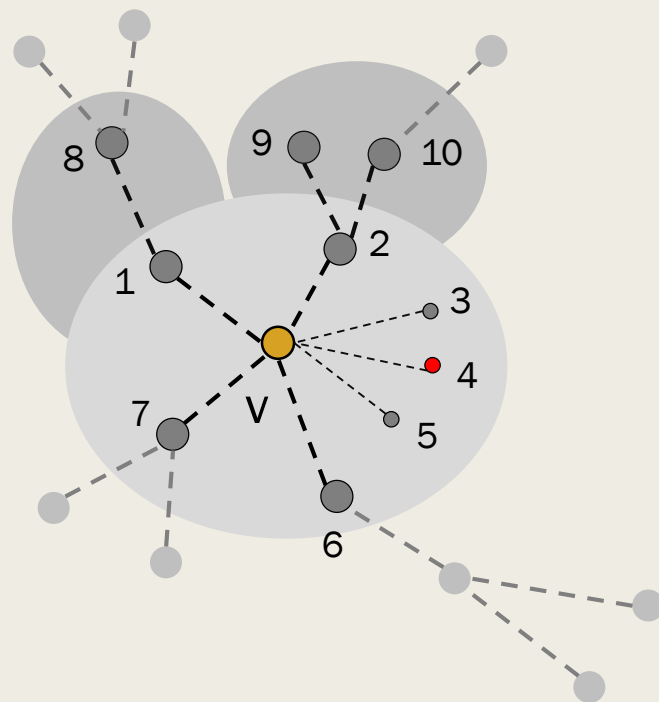
# BFS 廣度優先探索

- BFS 可以保證, 圖裡的點會依「與出發點間的距離」為順序被探索



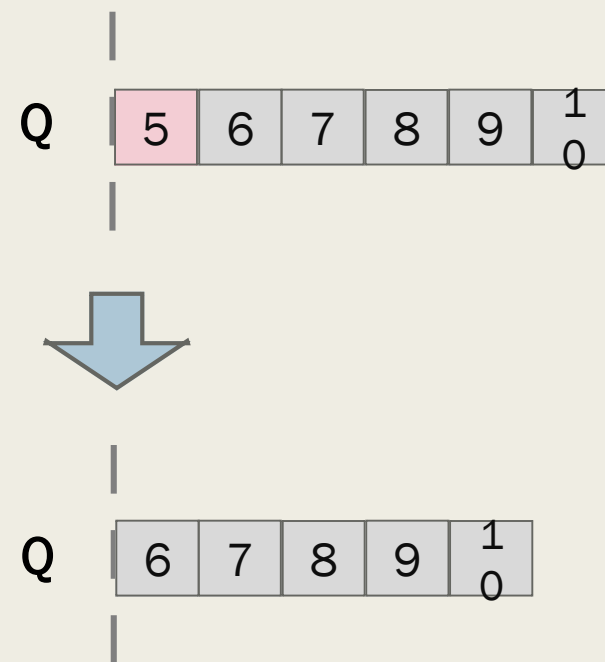
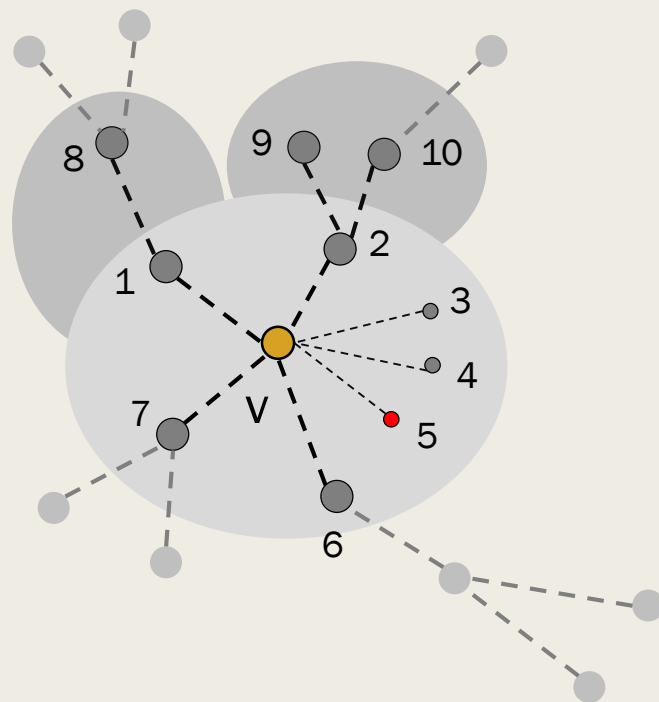
# BFS 廣度優先探索

- BFS 可以保證, 圖裡的點會依「與出發點間的距離」為順序被探索



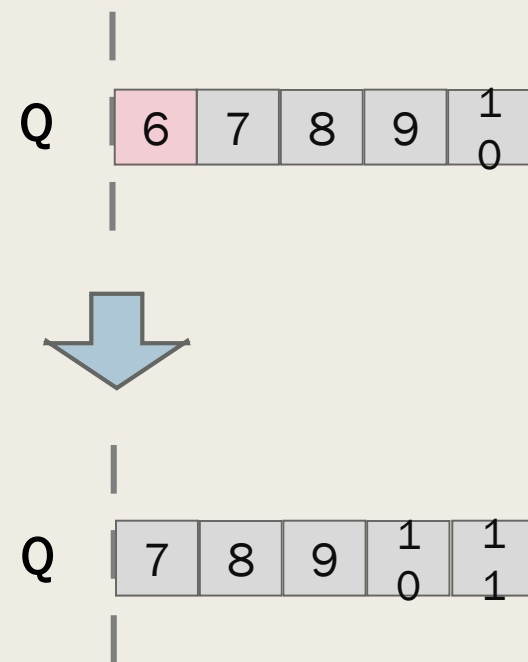
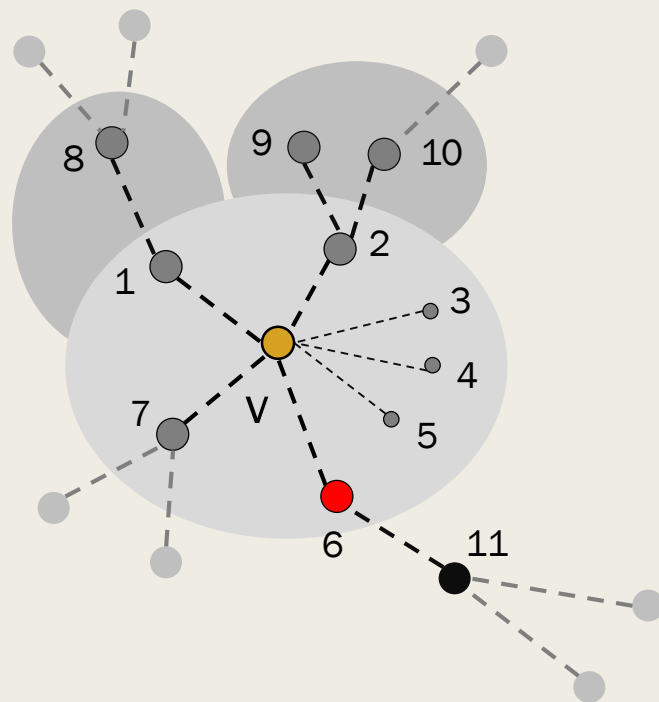
# BFS 廣度優先探索

- BFS 可以保證, 圖裡的點會依「與出發點間的距離」為順序被探索



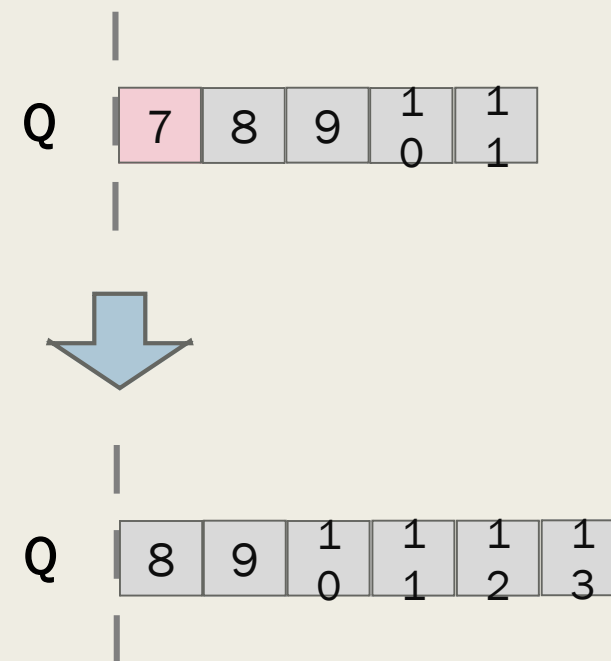
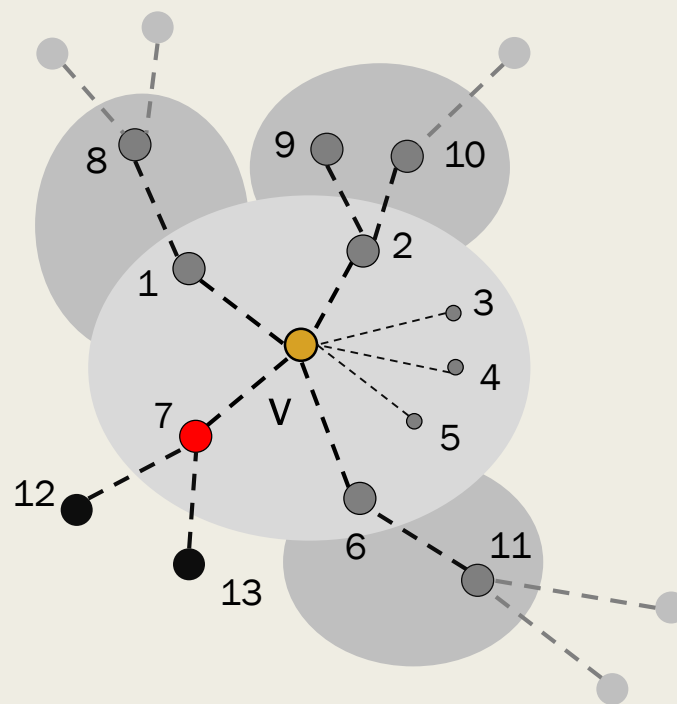
# BFS 廣度優先探索

- BFS 可以保證, 圖裡的點會依「與出發點間的距離」為順序被探索



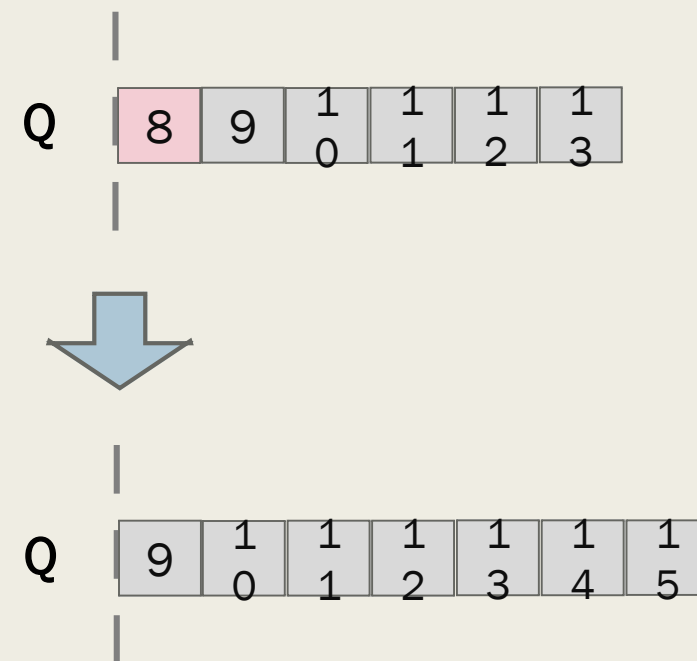
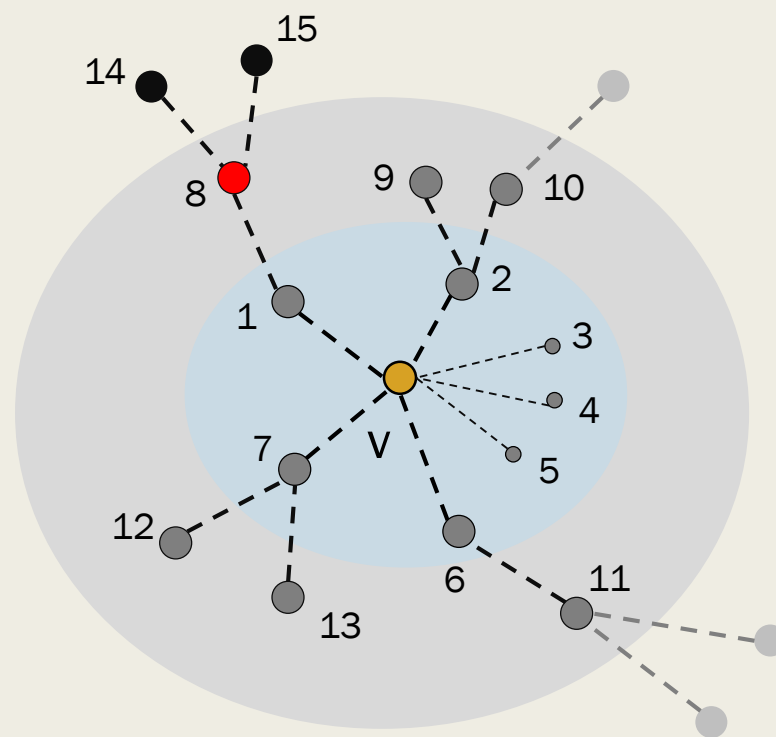
# BFS 廣度優先探索

- BFS 可以保證, 圖裡的點會依「與出發點間的距離」為順序被探索



# BFS 廣度優先探索

- BFS 可以保證, 圖裡的點會依「與出發點間的距離」為順序被探索





# BFS 廣度優先探索

■ BFS 探索的時間為  $O(|V|+|E|)$ , 與圖的點與邊數成正比。

- 每個點只會被放進  $Q$  裡一次、從  $Q$  裡被移除一次。
- 每條邊, 在過程裡只會被考慮兩次。

while 迴圈執行的次數為  $O(|V|)$

while 迴圈內的 for 迴圈總執行時間為  $O(|E|)$