

1.9 子集合加總問題

子集合加總 (subset sum) 問題的描述如下: 給定包含 n 個數值的集合 $\{v_1, v_2, \dots, v_n\}$ 及一特定數值 c , 決定是否存在元素加總值為 c 的子集合。

子集合加總問題可以使用動態規劃演算法解決, 其解題概念類似 0/1 背包動態規劃演算法, 說明如下。

定義 $v[i, w]$ 為 $\{v_1, v_2, \dots, v_i\}$ 的子集合中, 加總最大但不超過 w 的數值。因此, 若我們可以驗證 $v[n, c] = c$, 則可判斷存在一個子集合元素數值和為 c , 演算法回傳 True, 否則回傳 False。

Algorithm 子集合加總演算法

Input: 包含 n 個數值的集合 $\{v_1, v_2, \dots, v_n\}$ 及一特定數值 c

Output: True 或 False, 分別代表“存在”或“不存在”元素加總值為 c 的子集合

```
1: for  $w \leftarrow 0$  to  $c$  do
2:    $v[0, w] \leftarrow 0$ 
3: for  $i \leftarrow 1$  to  $n$  do
4:   for  $w \leftarrow 0$  to  $c$  do
5:     if  $v_i \leq w$  then
6:        $v[i, w] \leftarrow \max(v[i-1, w], v_i + v[i-1, w-v_i])$ 
7:     else
8:        $v[i, w] = v[i-1, w]$ 
9: if  $v[n, c] = c$  then
10:  return True
11: else
12:  return False
```

Sample input

3

3 7 10 5 2 6 17 56;33

14 28 72 69 39 5 11 3;238

323 452 371 498 167 84 595 999;2873

Sample output

True

True

False