

姓名:李展緯

學號:40747044s

演算法 hw3 handwritten

1.

(1)

Write down the state transition(recursive relation), and tabulate(draw a table) with the given items by using the transition you wrote. Please label the meaning of the column and the row in your table.

i\w	1	2	3	4	5	6
1	0	4	4	4	4	4
2	0	4	4	4	6	6
3	2	4	6	6	6	8
4	2	4	6	6	6	8

橫列:容量為 w 的袋子所能裝的物品最高價值

直行:編號為 i 的物品

$$V(i, w) = \max(v[i-1, w], v[i-1, w-w(i)] + v(i))$$

(2)

$$dp(i, j) = p(i) + (\text{sum}(i+1, j) - dp(i+1, j))$$

$$dp(i, j) = p(j) + (\text{sum}(i, j-1) - dp(i, j-1))$$

上述兩種情況中，因為對手一定拿最多的那一堆，因此用 i~j 的總和

$$P_1, \dots, P_6 = 2, 8, 3, 7, 5, 3$$

$$dp(i, j) =$$

{

$$dp(i, j) = p(i) \quad \text{if } i = j$$

$$dp(i, j).first = dp(j, j).first + a(i, j+1).second, dp(i, j).second = dp(i, j+1).first$$

$$\text{if } dp(i, j) dp(j, j).first + a(i-1, j).second > dp(i, i) dp(j, j).first + a(i-1, j).second$$

$$dp(i, j).first = dp(i, i).first + a(i-1, j).second, dp(i, j).second = dp(i-1, j).first$$

$$\text{if } dp(i, j) dp(j, j).first + a(i-1, j).second < dp(i, i) dp(j, j).first + a(i-1, j).second$$

}

$dp(i,j).first$ 為在 $p_i \sim p_j$ 中先手可拿到的數量

$dp(i,j).second$ 為在 $p_i \sim p_j$ 中後手可拿到的數量

$i \setminus j$	2	8	3	7	5	3
2	(2,0)	(0,0)	(0,0)	(0,0)	(0,0)	(0,0)
8	(8,2)	(8,0)	(0,0)	(0,0)	(0,0)	(0,0)
3	(5,8)	(8,3)	(3,0)	(0,0)	(0,0)	(0,0)
7	(15,5)	(11,7)	(7,3)	(7,0)	(0,0)	(0,0)
5	(10,15)	(15,8)	(8,7)	(7,5)	(5,0)	(0,0)
3	(18,10)	(16,10)	(10,8)	(10,5)	(5,3)	(3,0)

2.

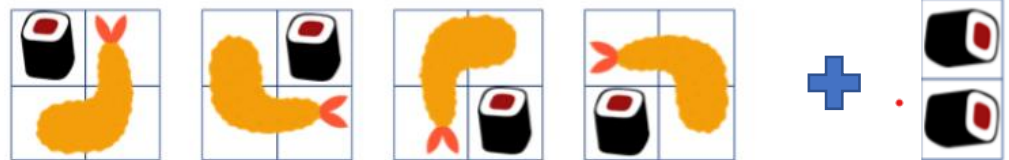
(1)

$dp(1) = 1$

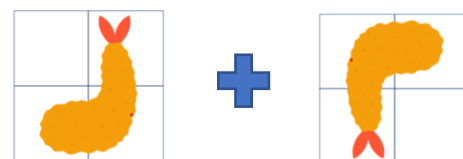


$dp(3) = 11$

8 種組合



2 種組合



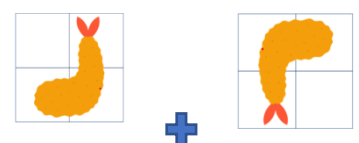
1 種組合



(2)

No

當 $dp = 3$ 時，此方程式不成立，因為有情況未被包含進去 如:



$$(3) dp(4) = dp(1) + dp(3) + dp(3) + dp(1) + dp(2) + dp(2)$$

$$(4) dp(5) = dp(4) + dp(1) + dp(1) + dp(4)$$

$$dp(5) = dp(4) + dp(1) + dp(1) + dp(4) = 2(dp(4) + dp(1)) = 2(dp(1) + dp(2) + dp(3) + dp(4))$$

$$dp(n) = 2(dp(n-1), \dots, dp(1))$$

3.

(1) 因為有 M 格，而每格只有 1 或 0 兩種選擇，因此為 2^M

(2) $2^M - 1 \sim 0$

(3) if (P&S != 0) 則可知道上下兩行必有相鄰的西瓜

(4)