

- Environment:

OS: Windows

Compiler: gcc/g++ version: 11.2.0

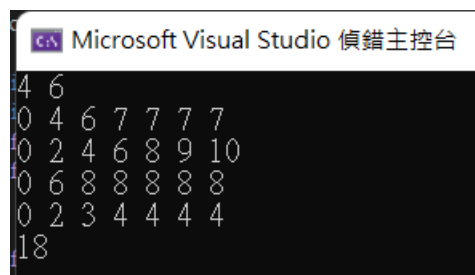
IDE: Visual Studio (version: 16.11.5)

可以用 cmd 開啟後執行，直接輸入 input



```
C:\Users\asd91\Desktop>.\109350008.exe
4 6
0 4 6 7 7 7 7
0 2 4 6 8 9 10
0 6 8 8 8 8 8
0 2 3 4 4 4 4
18
```

也可以用 visual studio 開啟編譯後直接輸入 input



```
Microsoft Visual Studio 偵錯主控台
4 6
0 4 6 7 7 7 7
0 2 4 6 8 9 10
0 6 8 8 8 8 8
0 2 3 4 4 4 4
18
```

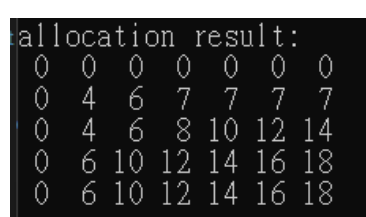
- Result:

- ◆ 方法:

設 $dp[i][j]$ 表示到第 i 個 project 以及第 j 個 resources 下所得的分配最大利益值，其中 i 代表到第幾個 project ($0 \leq i \leq n$), j 代表到第幾個資源 ($0 \leq j \leq m$)。若是 $i = 0$ 則 $dp[i][j] = 0$ (都沒有 project)。接下來依序下列步驟找出 $dp[n][m]$:

1. 先設 $dp[i][j] = dp[i - 1][j]$ (到 $i-1$ 個 project 時所分配下來的最大利益值)
2. 再來比較 $dp[i][j]$ 以及 $dp[i - 1][j - k] + profit[i][k]$ (第 $i-1$ 個 project 中第 $j-k$ 個 resources 的最佳利益數加上第 i 個 project 的第 k 個 profit), 再把最大值存入 $dp[i][j]$, 持續比較直到 $k = j$ 為止 ($0 \leq k \leq j$)。
3. 答案為 $dp[n][m]$

*產生的 dp table:



```
allocation result:
0 0 0 0 0 0 0
0 4 6 7 7 7 7
0 4 6 8 10 12 14
0 6 10 12 14 16 18
0 6 10 12 14 16 18
```

◆ Complexity:

$$n * (1 + 2 + 3 + \dots + m) = n * \frac{m * (m + 1)}{2} = \frac{n * m^2}{2} + \frac{n * m}{2} = O(n * m^2)$$