**演算法程式作業二**

110403518 林晉宇

1. **UVA 1257 Your ways**

本次作業題目為uva 1257 your ways，並且障礙物為獨立，使用兩種方法實作: 組合學及動態規劃，透過比較兩者的時間複雜度，產生測資並以圖表呈現兩者的時間差距。題目輸入包含h,w(地圖大小 <=1000)，k(天數 <=10000),q(障礙物數量)。

1. **組合學**

透過教授上課簡報第六頁的方法去實作，path[i][j]代表從起點到座標(i,j)的方法數，**遞迴式如下:**

這樣的**時間複雜度**為**O(KHW)**，總共K天，每一天都要花HW的時間去重新建表。

1. **動態規劃**

我參考李品雋同學簡報的方法來實作:

1. block[]儲存障礙物座標(依照離終點由遠到近排序)。
2. sub[i]代表障礙物i左下角涵蓋的所有障礙物。
3. dp[i]代表從起點到block[i]起始位置並且沒有經過sub[i]任一障礙物的路徑數量。
4. p[(x,y)]代表在無障礙物下，起點到該座標的路徑數量。

**遞迴式:**

**Pseudo code:**

***//填sub***

***for(i = 0~q)***

***{***

***tmp = [];***

***for(j = 0~i-1)***

***if(block[i] dominate block[j])***

***tmp.push\_back(j)***

***sub[i] = tmp;***

***}***

***//算dp***

***int path(i)***

***{***

***q\_path=0;***

***for(j:sub[i])***

***q\_path+=dp[j]\*p[block[i].s-block[t].t];***

***return p[block[i].s] - q\_path***

***}***

***dp[0] = p[block[0]]; //初始化***

***for(i = 0~q)***

***dp[i]=path(i); //推算dp***

**時間複雜度: O(HW**+**KQ^2)**，一開始先用加法原理建出p[]，總共k天，每天需要:

1. 排序障礙物，O(QlogQ)
2. 填sub[]，O(Q^2)
3. 推算dp[]，O(Q)

則每天要花O(Q^2)的時間。

1. **設計測資**

經過前面的實作及時間的推算，方法一為O(KHW)，方法二為O(HW+KQ^2)

而H,W (<=1000)，K(<=10000)，Q(<=100)，我認為最能看出兩者差異的為變數K，所以在設計測資方面，H,W皆設為最大值，，K則從1慢慢遞增。

1. **時間比較**
2. **結論**