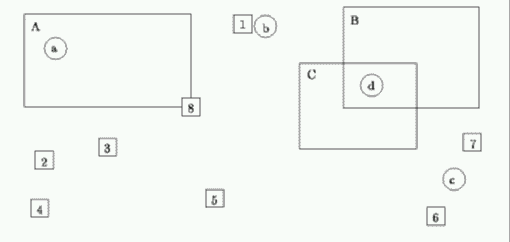
# Q142: Mouse Clicks

在windows作業系統中，你可以在桌面（desktop）上看到一些圖像（icon）或區域（工作視窗），他們在桌面都定義出一定的區域。當你按下滑鼠的按鍵時，系統必須知道現在滑鼠游標在哪裡，哪一個icon或工作視窗被選到了。在這個問題中，你要回答當滑鼠按鍵被按下去時，那一個icon或區域被選中了（按在區域的邊上也算選中）。如果滑鼠游標正好沒有在任何icon或工作視窗上，則距離游標最近的icon會被選上（有可能會同時選上2個以上相同距離且最近的icon）。舉例說明，以下圖表示電腦螢幕上桌面的配置關係圖：



當你在a的位置按下滑鼠按鍵時，A區域會被選中。當你在b的位置按下滑鼠按鍵時，1這個icon會被選中。當你在c的位置按下滑鼠按鍵時，6和7這2個icon會同時被選中，因為他們與c的距離相同。但是當你在 d的位置按下滑鼠按鍵時，事情會變的含糊不清，因為這個位置同時位於B與C的區域中。要解決這個問題，我們得看是那個區域在比較上方，然後系統會選擇比較上方的區域。由於區域是大寫英文字母來表示，同時也是區域出現的次序，所以C區域比B區域晚出現，也就是說從桌面的配置來看，C區域會蓋住B區域的某部分。因此，當滑鼠按在d時，系統會選擇C區域。請注意：區域總是會蓋住icon，所已被遮蓋住的icon不需要被考慮。還有，左上角的座標為(0,0)

寫一個程式讀入一連串icon及工作視窗的資料，然後再讀入滑鼠按鍵時的游標位置資料，回答選中哪些物件。螢幕桌面可視為X-Y平面，所有的座標值均介於0到499，並且你可以假設icon和工作視窗均是整個位於螢幕桌面上。你的程式必須依icon讀入的順序從1開始依次來編號。並且依工作視窗讀入的順序從A開始依次來編號。

**Input**

輸入包含許多列，每列的第一個字元表達了不同的資料型態。I代表icon，R代表工作視窗，M代表按下滑鼠按鍵。I和R可能混著出現，但是M一定是在輸入的最後面。  
I之後一定跟著2個整數，代表此icon的中心座標。R之後一定跟著4個整數，代表此工作視窗左上角及右下角的座標。M之後一定跟個2個整數，代表滑鼠按鍵被按下時滑鼠游標所在的座標位置。至少一定有一個可見的icon，並且不會有超過25個工作視窗及50個icon。輸入的最後一列僅含有一#，代表輸入結束。請參考Sample Input。

**Output**

對每一個滑鼠按鍵事件，輸出一列。如果是選中工作視窗，請輸出工作視窗的編號。如果是選中icon，請輸出icon的編號（長度3,靠右對齊）。若同時選中超過2個icon，則依編號由小到大輸出。請參考Sample Output。

**Sample Input**

I 216 28

R 22 19 170 102

I 40 150

I 96 138

I 36 193

R 305 13 425 103

I 191 184

I 387 200

R 266 63 370 140

I 419 134

I 170 102

M 50 50

M 236 30

M 403 167

M 330 83

#

**Sample Output**

A

1

6 7

C

程式碼

Imports System.IO

Public Class Form1

Dim fr As New FileInfo("input.txt")

Dim fw As New FileInfo("output.txt")

Dim sw As StreamWriter = fw.CreateText

Dim ii(2) As String

Dim n, a1(2), x, min As Integer 'n=輸入多少行資料

Private Sub Form1\_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load

If fr.Exists = False Then

End

ElseIf fw.Exists = False Then

fw.Create()

End If

Dim sr As StreamReader = fr.OpenText

Do

ii(n) = Trim(sr.ReadLine) '取出每行輸入資料

If ii(n) = "#" Then Exit Do

'以下3行指令皆為判斷I、R、M的指令分別有多少，好方便宣告陣列

If Microsoft.VisualBasic.Mid(ii(n), 1, 1) = "I" Then : a1(0) += 1

ElseIf Microsoft.VisualBasic.Mid(ii(n), 1, 1) = "R" Then : a1(1) += 1

ElseIf Microsoft.VisualBasic.Mid(ii(n), 1, 1) = "M" Then : a1(2) += 1

End If

n += 1 : ReDim Preserve ii(n)

Loop

n -= 1 : ReDim Preserve ii(n)

Dim aa1(a1(0) - 1, 1), aa2(a1(1) - 1, 3), aa3(a1(2) - 1, 1) As Integer '宣告I、R、M的陣列

a1(0) = 0 : a1(1) = 0 : a1(2) = 0

For i = 0 To n

x = -1

For j = 1 To Len(ii(i)) '分解每行的輸入

If Microsoft.VisualBasic.Mid(ii(i), 1, 1) = "I" Then

For k = j + 1 To Len(ii(i))

If Microsoft.VisualBasic.Mid(ii(i), k, 1) <> " " Then

aa1(a1(0), 0) &= Microsoft.VisualBasic.Mid(ii(i), k, 1)

If Microsoft.VisualBasic.Mid(ii(i), k + 1, 1) = " " Then

For l = k + 2 To Len(ii(i))

If Microsoft.VisualBasic.Mid(ii(i), l, 1) <> " " Then

aa1(a1(0), 1) &= Microsoft.VisualBasic.Mid(ii(i), l, 1)

If l = Len(ii(i)) Then a1(0) += 1 : GoTo 1

End If

Next

End If

End If

Next

ElseIf Microsoft.VisualBasic.Mid(ii(i), 1, 1) = "R" Then

For k = j + 1 To Len(ii(i))

If Microsoft.VisualBasic.Mid(ii(i), k, 1) <> " " Then

aa2(a1(1), x) &= Microsoft.VisualBasic.Mid(ii(i), k, 1)

If k = Len(ii(i)) Then a1(1) += 1 : GoTo 1

ElseIf Microsoft.VisualBasic.Mid(ii(i), k + 1, 1) <> " " Then

x += 1

End If

Next

ElseIf Microsoft.VisualBasic.Mid(ii(i), 1, 1) = "M" Then

For k = j + 1 To Len(ii(i))

If Microsoft.VisualBasic.Mid(ii(i), k, 1) <> " " Then

aa3(a1(2), 0) &= Microsoft.VisualBasic.Mid(ii(i), k, 1)

If Microsoft.VisualBasic.Mid(ii(i), k + 1, 1) = " " Then

For l = k + 2 To Len(ii(i))

If Microsoft.VisualBasic.Mid(ii(i), l, 1) <> " " Then

aa3(a1(2), 1) &= Microsoft.VisualBasic.Mid(ii(i), l, 1)

If l = Len(ii(i)) Then a1(2) += 1 : GoTo 1

End If

Next

End If

End If

Next

End If

Next

1: Next

Dim c(a1(0) - 1), b(2) As Integer

For j = 0 To a1(2) - 1 '每個點選地區

n = 0 : ReDim b(2), c(a1(0) - 1) : x = 0

For k = 0 To a1(1) - 1 '先找出它是否位於某些視窗中

If aa3(j, 0) > aa2(k, 0) And aa3(j, 1) > aa2(k, 1) And aa3(j, 0) < aa2(k, 2) And aa3(j, 1) < aa2(k, 3) Then

b(n) = k + 65 : x += 1

n += 1 : ReDim Preserve b(n)

End If

If k = a1(1) - 1 And x <> 0 Then

min = b(0)

For l = 0 To n - 1 '若有重複堆疊,則判斷哪個視窗是最後出現的

If min < b(l) Then min = b(l)

Next

sw.WriteLine(Chr(min)) : GoTo 2 '輸出

End If

Next

'----------------------------------------------------------------------

For k = 0 To a1(0) - 1 '若無在視窗中,則先求出他離所有icon的距離的2次方

c(k) = (aa1(k, 0) - aa3(j, 0)) ^ 2 + (aa1(k, 1) - aa3(j, 1)) ^ 2

Next

min = c(0) : x = 0

For k = 0 To a1(0) - 1 '比對差,找出離最近得icon

If min > c(k) Then min = c(k) : x = k

Next

n = 0

For k = 0 To a1(0) - 1 '找出是否有一樣近的icon

If min = c(k) Then

b(n) = k

n += 1 : ReDim Preserve b(n)

End If

Next

Array.Sort(b) '小到大排序

For k = 1 To n '輸出

sw.Write(Space(3 - Len(CStr(b(k) + 1))) & b(k) + 1)

Next

sw.WriteLine()

2: Next

sw.Flush() : sw.Close() : End

End Sub

End Class