

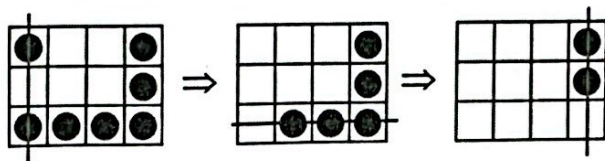
問題描述

為慶祝師大百年校慶，林老師發明了「縱橫殺棋對抗賽」的遊戲，讓玩家們易玩難精。給定一個 n 行 m 列的矩形棋盤，其中有些格子放置有棋子，兩個玩家輪流玩一個「縱橫殺棋對抗賽」的遊戲。輪到走步的玩家可選擇至少有一顆棋子的縱向或橫向的任一行或任一列，將此行或此列的所有棋子取走。當所有棋子均被取光了，遊戲就結束。這時就看哪一方取走的棋子總數較多，以此決定輸、贏、或和棋。

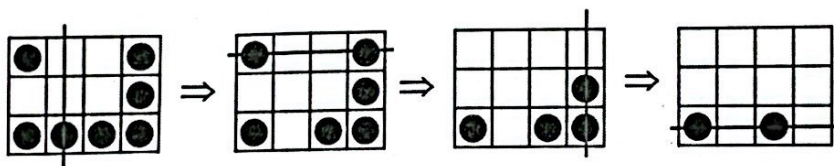
下圖是一個 4 行 3 列的棋盤可能的過程例子。最上方顯示先手方取走第一行兩顆棋子，接著後手方取走第 3 列的 3 顆棋子，最後先手方取走第 4 行剩下的兩顆棋子。此局先手得 2+2 子，後手得 3 子。故先手贏了 1 子。

反過來，若下錯，如中間部份所示，先手得 1+2 子，後手得 2+2 子。故先手輸了 1 子。

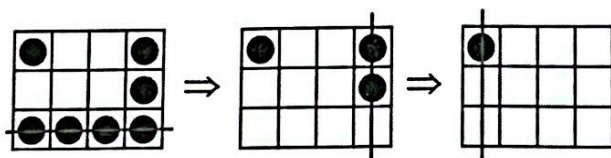
當然，為了贏且贏更多，先手方可改取走第 3 列的 4 顆棋子，那麼先手得 4+1 子，後手得 2 子。故先手贏 3 子。這也是先手方及後手方均為最佳走步的結果。



⇒ 先手得 2+2 子，後手得 3 子。故先手贏 1 子。



⇒ 先手得 1+2 子，後手得 2+2 子。故先手輸 1 子。



⇒ 先手得 4+1 子，後手得 2 子。故先手贏 3 子。

現在要請你寫一個程式，來針對初始盤面，找出先手方的一個最佳走步即可。也就是若能贏，則贏最多。若不能贏，則需為輸最少。

輸入說明

輸入資料為給定的初始盤面，一律存在 c:\input.txt 或自訂位置。第一行有兩個整數 n 及 m ，依序表示棋盤的列數與行數，其中 $1 \leq n, m \leq 8$ 。接下來有 n 行，每行有 m 個數字，數字包含 {0, 1} 並且數字之間以一個空白隔開，表示初始盤面每一列的格子中棋子擺放狀況。若為 1，表示該格有棋子；若為 0，表示棋盤上該格沒有棋子。注意：我們給定的盤面中至少會有一顆棋子。

輸出說明

輸出資料一律存在 c:\output.txt 或自訂位置。請輸出三行：第一行，含有若干個數字，中間以空白隔開，表示先手方的一個最佳走步的行或列的位置，分別用 Row # 或 Column # 及 1~8 來表示這一個位置。第二行則請輸出先手方最後會贏或輸多少子，正數表示贏多少子，負數表示會輸多少子，0 則表示是和棋。第三行輸出總執行時間，可以設法透過時間函數來取得。

範例

輸入範例一	輸出範例一
3 4	Row #: 3
1 0 0 1	+3 points
0 0 0 1	Total run time = 0.08 seconds.
1 1 1 1	
輸入範例二	輸出範例二
3 4	Row #: 1
1 1 1 1	0 points
0 0 0 0	Total run time = 0.03 seconds.
1 1 1 1	
輸入範例二	輸出範例二
4 8	Column #: 4
1 1 1 1 0 0 0 0	4 points
0 0 0 1 1 1 1 1	Total run time = 0.6 seconds.
0 0 0 1 1 1 1 1	
1 1 1 1 0 0 0 0	

此作業將要求你使用 alpha-beta pruning 演算法來開發一個下此棋的程式。此作業要求能下棋即可，但愈快、愈聰明、介面愈美觀的程式分數會愈高。

作業報告：將程式之原始碼、執行檔、整個專案及作業報告上傳到師大 Moodle 數位學習平台 (<http://moodle.ntnu.edu.tw/>)。作業報告統一繳交 PDF 檔。不要用其它格式。請務必確認系統已收到你的資料。

- 一、請說明如何執行你的程式，並詳細說明你所使用之機器軟硬體規格、作業系統、開發軟體版本相關資訊以及你為何選擇這樣的規格。另外請提供你的連絡電話，以便不時之需。（這部份約佔 10% 的分數）
- 二、原始碼中應加註解，請說明如何執行這個程式。（約佔 10% 的分數）
- 三、首先你先設法製作測試用的輸入檔至少 5 個。盤面尺寸需有大有小，解題難度也不同，由你決定。請說明你如何製作這些測試用的輸入檔。（約佔 10% 的分數。提示：可考慮用亂數產生）
- 四、說明程式中你使用甚麼方法、甚麼資料結構、甚麼技術（操練要項）來解決這個問題，並請你說明你測試一些盤面時的表現如何。請你用一些例子輔助說明，並說明你的程式具有那些功能及優點。（約佔 50% 的分數）
- 五、說明你的參考文獻及網站及參考了那些部份用於實作中。（約 10% 的分數）
- 六、請說明你發展此程式所碰到的一些狀況及困難。（這部份約佔 10% 的分數）
- 七、額外加分：如果你能利用一些現有的或自己想出來的進階技術，來得出一些成果，則另有加分。

注意事項：

- 如果你有做美美的下棋人機介面，可和人下棋，則介面可完全自由設計，但至少應在畫面上提示給玩者，讓他知道如何操作。另外也至少要有配合我們作業的要求：由我們給定的初始盤面 input.txt 開始下，並且將先手方的第一步最佳走步輸出到 output.txt，以方便批改。
- 此作業要求能下棋即可，但愈快、愈聰明、介面愈美觀的程式分數會愈高。
- 本課程允許遲交作業（三天內），遲交的話，一天內約扣 2 分，二天內約扣 4 分，三天內約扣 6 分。
- 提醒：報告及程式應親自為之。報告如有抄襲同學之狀況（不管被抄襲或抄襲別人），一律依校規送校方處理。請你也要保護好你的智財權，別讓別人不勞而獲。因此作業請勿抄襲同學的作業。根據以往的經驗，非常容易被發現~。（因為修課人數不是大數據的規模）
- 作業報告著重於解決這個問題的構想、理由，還有實驗設計以及結果說明、分析的完整性，即使實驗數據不好看、測試效能不佳，只要有完整的結果分析、或是說明有獨到的見解都能拿高分。換句話說，報告的評分重點在你對於作業的用心程度，自己構思所佔的部份是否足夠，而不僅僅是測試效能、實驗數據的美觀。