

Problem 1. 時間轉換

(Time Limit: 2 seconds)

問題描述：

隨著台灣經濟的起飛，越來越多人有能力出國度假，此時時差上面的轉換就格外的重要。請寫一個可轉化台灣、美國、日本、英國的對照表，不須考慮日光節約時間且不考慮日期。

輸入說明：

第一列輸入一個正整數 n ($0 < n < 10$)。其後有 n 列，每一列代表一個想要轉換的時間，每一列之資料依序為想轉換的時間、目前所在的國家、想轉換的國家。時間為 24 小時制 ($0 - 23$)，以 4 個阿拉伯數字表示。例如下午六時表示 1800。國家代號如下：台灣用 TW；美國分美西 USW，美東 USE，美中 USC；日本用 JA；英國用 UK；TW 晚 JA 1 小時；TW 早 USE 12 小時；TW 早 USC 13 小時；TW 早 USW 14 小時；TW 早 UK 8 小時

輸出說明：

每列之資料用空白分隔，每列資料有轉換過的時間與想轉換的國家，共 n 筆，最後必須有換行字元。

範例：

Sample Input:	Sample Output:
2	2000 JA
1900 TW JA	1100 UK
0600 USC UK	

Problem 2. 邊緣偵測

(Time Limit: 2 seconds)

問題描述：

一個二元圖形通常用一個二維陣列表示，其中的值為 0 或 1，即為淺色（以 1 表）圖形在深色（以 0 表）的背景中。在分析圖形時往往需要找出其邊緣，稱之為邊緣偵測。所謂”邊緣”即為一個 1 的 pixel 其上下左右至少有一個為 0。寫一個程式讀入二元圖形，輸出該圖之邊緣。

輸入說明：

第一行為一個正整數 N ($0 < N < 10$)，代表共有幾組測試資料。之後接下來有 N 組數據，每組第一行為兩個正整數 n 及 m ，表示二元圖形的大小為 $n \times m$ 的矩陣，之後 n 行每行有 m 個數字 ($1 \leq n, m \leq 100$ ，每個數字後面都有一個空格字元)，分別是 0 或 1，即為該組測試資料之二元圖形。

輸出說明：

將不是邊緣的點以底線 ‘_’ 表示，邊緣的點則以 0 表示，每組測資答案按照該組測資給予的 矩陣大小輸出於 n 行，每輸出一個點，後面即輸出空格。每組測試資料結果與前一組之間空一行，最後必須有換行字元。

範例：

Sample Input:	Sample Output:
2	-----
5 7	__ 0 0 0 __
0 0 0 0 0 0 0	_ 0 _ _ 0 _
0 0 1 1 1 0 0	__ 0 0 0 __
0 1 1 1 1 1 0	-----
0 0 1 1 1 0 0	
0 0 0 0 0 0 0	-----
8 11	__ 0 0 _ _ 0 0 __
0 0 0 0 0 0 0 0 0 0 0 0	_ 0 _ _ 0 _ 0 _ _ 0 _
0 0 1 1 0 0 0 1 1 0 0	__ 0 _ _ 0 _ _ 0 _
0 1 1 1 1 0 1 1 1 1 0	__ _ 0 _ _ 0 _ _
0 0 1 1 1 1 1 1 1 0 0	__ _ _ 0 _ 0 _ _
0 0 0 1 1 1 1 1 0 0 0	__ _ _ _ 0 _ _ _
0 0 0 0 1 1 1 0 0 0 0	-----
0 0 0 0 0 1 0 0 0 0 0	
0 0 0 0 0 0 0 0 0 0 0	

Problem 3. 遞迴走迷宮

(Time Limit: 2 seconds)

問題描述：

請運用遞迴程式的技術，找出一條迷宮的路徑。迷宮由 $N \times N$ 之二維陣列以 1 和 0 組成，1 代表障礙物，0 代表通道， N 為正整數($N < 15$)。入口為 (1, 1)，出口為 ($N-2, N-2$)。輸入迷宮的路徑只有唯一解。

輸入說明：

輸入檔案由 $N \times N$ 之 0 與 1 組成，1 代表障礙物，0 代表通道，迷宮周圍均為 1 組成，每一個字元以空白分隔。

	資料	意義
第一筆	0 與 1 組成 N 個字元，兩兩以空白間格	1 代表障礙物，0 代表通道
第二筆	0 與 1 組成 N 個字元，兩兩以空白間格	1 代表障礙物，0 代表通道
...	0 與 1 組成 N 個字元，兩兩以空白間格	1 代表障礙物，0 代表通道
第 N 筆	0 與 1 組成 N 個字元，兩兩以空白間格	1 代表障礙物，0 代表通道

輸出說明：

	資料	意義
第一筆	0、1 與 # 組成 N 個字元，兩兩以空白間格	1 代表障礙物，0 代表通道，# 代表找到之路徑
第二筆	0、1 與 # 組成 N 個字元，兩兩以空白間格	1 代表障礙物，0 代表通道，# 代表找到之路徑
...	0、1 與 # 組成 N 個字元，兩兩以空白間格	1 代表障礙物，0 代表通道，# 代表找到之路徑
第 N 筆	0、1 與 # 組成 N 個字元，兩兩以空白間格	1 代表障礙物，0 代表通道，# 代表找到之路徑

最後必須有換行字元。

範例：

Sample Input:	Sample Output:
111111111111	111111111111
100000000001	1###00000001
111011011111	111#11011111
111001111111	111##1111111
111100000001	1111##000001
111110111111	11111#111111
100000000001	10000###0001
111010101111	1110101#1111
111010100001	1110101###1
11101010101	111010101#1
111111111111	111111111111

Problem 4. 網路節點中介值

(Time Limit: 2 seconds)

問題描述：

在圖（網路）中「節點中介值」的定義包括三個節點：一個起始節點，一個目標節點，再加上一個要計算「節點中介值」的節點。「節點中介值」為由起始節點到目標節點的所有「網路最短路徑」的個數除這些最短路徑中通過要計算「節點中介值」節點的數目。假設由起始節點到目標節點共有 10 條最短路徑，其中有 2 條通過某節點，那麼其「節點中介值」則為 $2/10=0.2$ 。給予一代表無向連通圖的鄰接矩陣並指定起始節點，目標節點與要計算的節點，請計算此節點的「節點中介值」。

輸入說明：

輸入第一行是測資筆數，每筆測資格式如下。

每筆輸入分為兩部份，第一部份只有一行，此行中有四個用逗號分隔開的介於 1 到 99 的數字，第一個數字 N 代表要計算的圖中節點數目。第二個數字 S 與第三個數字 T 分別表示起始節點與目標節點。第四個數字 V 則是要計算「節點中介值」的節點編號。第二部份是鄰接矩陣，總共列數剛好有節點個列，每一列中有節點個數個由空格分開的 0 或 1。第 i 列中的第 j 個 0 或 1 代表由節點 i 到節點 j 的連線是否存在（0 表示不存在，1 表示存在）。沒有自我迴圈連線表示第 i 列中的第 i 個元素值為 0。連線沒有方向性，因此鄰接矩陣為對稱矩陣。

輸出說明：

每筆測資在同一行輸出兩個數字，以一個空白間隔。先輸出 S 到 T 的最短路徑數，再輸出 N 在幾條 S 與 T 的最短路徑上，每行結尾都必須有換行字元。

範例：

Sample Input:	Sample Output:
1 9,1,5,2 0 1 0 1 0 0 0 0 1 0 1 0 1 0 0 0 0 1 0 0 0 1 0 0 1 0 0 0 1 0 1 0 0 1 0 1 0 1 0 1 0	2 1

001010001	
000100010	
000010101	
000001010	

Problem 5. Metric next-to-shortest path

(Time Limit: 2 seconds)

Problem Description

In many applications, we need to find a shortest path between two nodes in a network. In this problem you are asked to find the length of a next-to-shortest path which is the minimum length path amongst all paths strictly longer than the shortest path. Finding the next-to-shortest path seems more complicated than the shortest path problem. But it remains polynomial time solvable. Here you are asked for a simpler version of the problem, for which the input is a metric. A metric can be thought of as an undirected complete graph whose weight satisfying the triangle inequality, i.e., $w(a,b) \leq w(a,c) + w(c,b)$ for any nodes a,b,c , in which $w(a,b)$ is the weight of the edge between a and b .

Input Format

The first line has an integer which indicates the number of test cases. The first line of each test case is an integer n , $1 < n < 100$, which is the number of nodes in this case. The next $n-1$ lines consist of the upper-triangle matrix of the edge lengths. Each weight is a positive integer less than 10000.

Output Format

For each case, print the length of the next-to-shortest path from node 0 to node $n-1$ in one line. Note that a path must be simple. That is, no node can appear twice in a path. If there is no such path, “-1” should be output.

Example

Sample Input:	Sample Output:
2 5 1 2 3 5 1 3 4 2 3 3 3	6 -1

1 2	
1	