

HW12-1. 簡易影像處理

在影像處理中，數位影像其實就是一個巨大的二維矩陣 (2D Matrix)，矩陣中的每一個數值代表該像素 (Pixel) 的顏色或灰階值。

「轉置 (Transpose)」是一個非常基礎的操作。轉置矩陣 A^T 的定義是將原矩陣 A 的列 (Row) 與行 (Column) 互換，原本位於座標 (i,j) 的元素，轉置後，會移動到座標 (j,i)。

請撰寫一個程式，讀入一個 $N \times N$ 的整數矩陣，並輸出它的轉置矩陣。

Input

- 第一行輸入一個整數 N ($1 \leq N \leq 100$)，代表圖片矩陣的長寬。
- 接下來有 N 行，每一行包含 N 個整數，代表原始圖片的像素資料。

Output

- 輸出處理完成後的 $N \times N$ 矩陣。
- 數字之間請以一個空白隔開，每一列結尾請換行。

有無空格、大小寫、符號、換行請同學再多比對一下，建議直接複製範例測資的文字

Sample Input 1

3	1 4 7
1 2 3	2 5 8
4 5 6	3 6 9
7 8 9	

Sample Output 1

Sample Input 2

2	10 30
10 20	20 40
30 40	

Sample Output 2

HW12-2. 忙碌的助教

助教最近非常忙碌，行事曆上充滿了各種會議與待辦事項。每個事項都有一個「開始時間」與「結束時間」。

然而有些事項的時間是重疊的。例如：助教在 13:00~15:00 要開會，同時在 14:00~16:00 要改作業。這意味著助教從 13:00 一路忙到 16:00 是一段連續不可分割的忙碌時間。

為了簡化行事曆，請你利用 C++ 的 `vector` 與 `struct`，幫助教把所有**重疊的時間區間 合併 (Merge)**，整理出一份最簡潔的時間表，並計算助教總共忙了多久。

Input

- 第一行輸入一個整數 `N`，代表行事曆上有幾個事項。
- 接下來 `N` 行，每行包含兩個整數 `start` 與 `end` (`start < end`)，代表該事項的開始與結束時間。

注意：

1. 輸入的時間區間未必是依照順序排列的（記得排序每個時段）。
2. 輸入採用二十四小時制(0-23)，測資以小時為單位，不考慮分鐘的計算，且**不存在**工作至跨日以及不用工作的情況。

Output

- 第一部分：請依序印出合併後的所有時間區間（格式：`start end`），每行一個。
- 第二部分：請印出 `Total Duration: X`，其中 `X` 為所有合併後區間的總長度。

有無空格、大小寫、符號、換行請同學再多比對一下，建議直接複製範例測資的文字

Sample Input 1

4	1 6
1 3	8 10
8 10	15 18
2 6	Total Duration: 10
15 18	

Sample Output 1**Sample Input 2**

2	1 4
1 4	Total Duration: 3
2 3	

Sample Output 2**Sample Input 3**

2	1 5
1 3	Total Duration: 4
3 5	

Sample Output 3

繳交格式

1. 上傳內容須為 **.cpp** 檔
 - 檔名為: 班級 XX-學號
(XX:第幾份作業, i.e. A01-114502000)
2. 請針對程式內的重要功能進行「**適當註解**」, 簡單說明該段程式碼的作用, 註解的部分會納入評分考量, 請勿隨意註解
3. 記得在 **Online Judge** 系統上測試, 若在正常繳交期限前未有測試紀錄, 將扣該次作業 10 分
4. 若檔名錯誤, 扣該次作業 5 分

繳交期限

- **2025/11/28 23:59** 前將檔案上傳到 **ee-class** 上

遲交期限

- **2025/11/30 23:59** 前將檔案上傳到 **ee-class** 上
- 該次作業分數打八折
- 遲交期限過後不再開放繳交作業, 該次以 0 分計算