



2026

NTUCPC Programming Contest



Jane Street



msi

決賽

A. 裸題

Problem ID: mst

裸題，就是沒有包裝，照做即可的題目。

這題，是一個裸題。

相信大家知道什麼是最小生成樹。對於一張 n 點 m 邊的邊帶權的圖，他的最小生成樹是邊權總和最小的、所有點連通且無環之子圖。子圖的定義是移除若干條邊（可以不移除邊）之後可以獲得的圖。

現在，有一張 $n = 1000$ 個點的完全圖，也就是任兩點之間恰有一條邊的圖，他總共有 $\binom{1000}{2} = 499500$ 條邊，這張圖的邊權是 1 到 $\binom{n}{2}$ 這 $\binom{n}{2}$ 個數字的**隨機排列**，你想要找到這張圖的最小生成樹。其中， $\binom{n}{2} = \frac{n(n-1)}{2}$ 。

但是，因為直接出最小生成樹出題者會被罵，所以我們並不會直接提供你每條邊的邊權，你必須透過問問題來找出這棵樹的最小生成樹。當然，因為這題是裸題，所以你能問的問題也很單純，就只是比較兩條邊的邊權誰比較小，我們最多可以回答你 749250 個問題。

當然，限制不只如此。因為 flush 很慢，一問一答很沒效率，所以你可以一次問好幾個問題，然後我們會一次回答你這些問題的答案，我們說這樣叫作「一輪詢問」。當然的，我們會限制你詢問的輪數，總共最多只能做 15000 輪詢問。

更精準地說，假設你總共進行了 $k \leq 15000$ 輪詢問，在第 i 輪詢問中，你要先選一個**正整數** S_i ，代表在這輪詢問你要問 S_i 個問題，然後詢問 S_i 對邊，其中第 j 對邊記作 (u_j, v_j) 和 (x_j, y_j) ，代表你想要問這兩條邊誰的邊權比較小。在送出這輪全部的問題之後，judge 會告訴你每個問題的答案，也就是 S_i 個大小關係，其中第 j 個就代表你詢問的第 j 對邊邊權的大小關係。別忘了你問的問題總數也不能太多，所以你必須保證 $\sum_{i=1}^k S_i \leq 749250$ 。

最後，我們很好心的告訴你：本題的測資數量為 12 筆。

互動格式

當要進行一輪詢問時，請輸出一行「? S_i u_1 v_1 x_1 y_1 u_2 v_2 x_2 y_2 ... u_{S_i} v_{S_i} x_{S_i} y_{S_i} 」（不含引號），其中 S_i 代表你這輪要詢問 S_i 個問題，且第 j 個問題要比較邊權的兩條邊是 (u_j, v_j) 與 (x_j, y_j) 。每個問題的兩條邊應相異且 $u_j \neq v_j, x_j \neq y_j$ ，同時 S_i 需為正整數。當你輸出完這些數字後，記得要清空（flush）標準輸出（standard output）。

當系統收到你的詢問後，會回覆一個長度 S_i 的字串到你的標準輸入，其中字串的第 j 個字元為下列兩者其中一個：



2026

NTUCPC Programming Contest



Jane Street



msi

決賽

- $<$ 代表 (u_j, v_j) 邊權較小；
- $>$ 代表 (x_j, y_j) 邊權較小。

當你猜出這張圖的最小生成樹後，請輸出一行「! $u_1 v_1 u_2 v_2 \dots u_{n-1} v_{n-1}$ 」(不含引號)，其中 $(u_1, v_1), (u_2, v_2), (u_{n-1}, v_{n-1})$ 為這棵生成樹的 $n - 1$ 條邊。輸出完後，你的程式必須立刻結束 (exit)。

你最多只能詢問 749250 個問題，且最多只能詢問 15000 輪。注意到輸出答案不計入這個次數內。如果你的程式詢問了超過 749250 個問題，或詢問了超過 15000 輪，又或是違反了題目敘述中的規則或輸出格式，你的程式將會被強制中止，並且會得到 WRONG-ANSWER。

範圍限制

- $n = 1000$
- $m = \binom{1000}{2}$
- 保證圖的邊權為 1 到 m 這 m 個數字的隨機排列。

以下是一個完整的互動範例（參賽者的輸出在右側，而系統的輸出，也就是參賽者的輸入在左側）。在下面的範例中，我們會以四個點的完全圖做為範例，但注意到在真正的測試資料中，所有的 $n = 1000$ 。

Sample Input	Sample Output
><<	? 3 1 2 2 3 1 2 2 4 2 3 2 4
>	? 1 1 3 1 3
><	? 2 1 2 1 4 1 2 1 3
	! 1 2 2 3 1 4



以下是 C 程式 flush 的範例：

```
1 #include <stdio.h>
2 int main() {
3     printf("500\n");
4     fflush(stdout);
5 }
```

以下是 C++ 程式 flush 的範例：

```
1 #include <iostream>
2 int main() {
3     std::cout << "500\n";
4     std::cout << std::flush;
5 }
```

以下是 Python 程式 flush 的範例：

```
1 import sys
2 print(500)
3 sys.stdout.flush()
```

以下是 Java 程式 flush 的範例：

```
1 System.out.println(500);
2 System.out.flush();
```

以下是 Kotlin 程式 flush 的範例：

```
1 fun main() {
2     println(500);
3     System.out.flush();
4 }
```

以下是 Rust 程式 flush 的範例：

```
1 use std::io::{self, Write};
2 fn main() {
3     println!("500");
4     let _ = io::stdout().flush();
5 }
```



2026

NTUCPC Programming Contest



Jane Street



msi

決賽

This page is intentionally left blank.