

## D. 隨機並查集

Problem ID: unionfind

你知道並查集（Disjoint-set Union）嗎？並查集是一種可以處理「快速合併兩個集合」並且詢問「某個元素所在的集合」這樣操作的資料結構。具體來說，對於  $N$  個由 1 到  $N$  形成的集合

$$S = \{\{1\}, \{2\}, \dots, \{N\}\}$$

並查集可以支援下列兩種操作：

1. 合併：指定  $x$  與  $y$ ，合併元素  $x$  與元素  $y$  所屬的集合。
2. 查詢：指定  $x$  與  $y$ ，查詢元素  $x$  與元素  $y$  是否同屬一個集合。

小 B 最近在演算法課程上學到了這個資料結構。專精隨機演算法的他，馬上開始好奇這個資料結構的「隨機」所需的執行週期（cycle）數。具體來說，在程式開始執行前，每個元素都在不同的集合。程式會花一個週期隨機選取  $1 \leq x < y \leq N$  並判斷  $x$  與  $y$  是否位於同一個集合。若  $x$  與  $y$  位於同個集合，則程式不做任何額外的事。若  $x$  與  $y$  分別位於集合  $A$  與集合  $B$ ，小 B 的程式會額外花  $f(|A|, |B|)$  個週期將集合  $A$  與集合  $B$  合併。這個過程會持續到所有元素都位於同一個集合為止。

以下的虛擬碼模擬了小 B 的程式：

---

### Algorithm 1 Stochastic Union-Find

---

```
1: procedure Emulate
2:    $cycle \leftarrow 0$ 
3:    $S \leftarrow \{\{1\}, \{2\}, \dots, \{N\}\}$ 
4:   while  $|S| > 1$  do
5:      $cycle \leftarrow cycle + 1$ 
6:     randomly choose two distinct elements  $x$  and  $y$ 
7:     if  $x$  and  $y$  belongs to sets  $A \neq B$  then
8:        $cycle \leftarrow cycle + f(|A|, |B|)$ 
9:       merge  $A$  and  $B$ 
return  $cycle$ 
```

---

小 B 想知道他的程式執行所需週期數的期望值，請你設計另一個程式幫幫他吧！

## Input

輸入第一行有一個正整數  $N$ ，代表元素的個數。接著有  $N$  行，每行有  $N$  個非負整數，第  $i$  行的第  $j$  個代表  $f(i, j)$ 。保證  $f(i, j) = f(j, i)$  且對於所有  $i + j > N$ ，不可能有合併大小  $i$  與大小  $j$  的集合的操作，因此方便起見  $f(i, j) = 0$ 。

- $1 \leq N \leq 40$
- $0 \leq f(i, j) \leq 10^9$

## Output

輸出所需週期數的期望值。可以證明答案為有理數  $\frac{P}{Q}$ ，為了避免浮點數誤差，請輸出  $PQ^{-1} \pmod{10^9 + 7}$ 。其中  $Q^{-1}$  為  $Q$  在模  $10^9 + 7$  以下的乘法反元素，也就是  $Q \times Q^{-1} \equiv 1 \pmod{10^9 + 7}$ 。

## Note

範例測試資料一中，總共只有一個集合，因此程式在一開始就結束，並不花費任何週期。

範例測試資料二中，一開始有兩個集合。第一個週期有 100% 的機率選到元素 1 與 2，並額外花一週期將它們合併。總共需要兩個週期。

範例測試資料三中，期望所需週期數為  $\frac{P}{Q} = \frac{5}{2}$ ，可以自行驗證  $500000006 \times 2 \equiv 5 \pmod{10^9 + 7}$ 。

數字  $x$  在模  $10^9 + 7$  以下的乘法反元素為  $y \equiv x^{10^9+5} \pmod{10^9 + 7}$ 。

### Sample Input 1

1	
0	

### Sample Output 1

0
---

**Sample Input 2**

2 1 0 0 0	2
-----------------	---

**Sample Output 2****Sample Input 3**

3 0 0 0 0 0 0 0 0 0	500000006
------------------------------	-----------

**Sample Output 3****Sample Input 4**

10 7 7 3 4 2 7 6 7 7 0 7 5 4 8 10 4 7 6 0 0 3 4 5 1 4 4 3 0 0 0 4 8 1 6 8 3 0 0 0 0 2 10 4 8 4 0 0 0 0 0 7 4 4 3 0 0 0 0 0 0 6 7 3 0 0 0 0 0 0 0 7 6 0 0 0 0 0 0 0 0 7 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	979356198
--	-----------

**Sample Output 4**

*This page is intentionally left blank.*