

## 1. Ламана

**Максимальна оцінка: 100 балів**

**Обмеження на час: 1 сек.**

**Обмеження на пам'ять: 256 МБ**

**Вхідний файл: polyline.in**

**Вихідний файл: polyline.out**

**Програма: polyline.\***

### **Завдання**

На площині позначено кілька точок. Якщо це можливо, побудуйте замкнену ламану без самоперетинів, що проходить через усі ці точки і не має відмінних від них вершин.

### **Вхідні дані**

У першому рядку вхідного файлу задано кількість точок  $n$ ,  $3 \leq n < 2017$ . У  $n$  наступних рядках вказано по дві координати кожної точки: абсцису та ординату відповідно. Точок, що повторюються, немає. Всі координати — цілі числа, що за модулем не перевищують 10 000.

### **Вихідні дані**

Якщо побудувати потрібну замкнену ламану неможливо, слід вивести 0. В іншому разі треба вивести порядок, у якому задані точки розташовані на ламаній, тобто порядок обходу всіх цих точок, починаючи з довільного місця і в довільному напрямку (або за, або проти годинникової стрілки). Нумерація точок починається з одиниці. Ламаних, що задовольняють умову, може бути кілька. Достатньо знайти хоча б одну.

### **Приклад**

<b>polyline.in</b>	<b>polyline.out</b>
7	2 4 3 1 6 5 7
1 0	
-1 -1	
1 1	
-1 1	
0 0	
1 -1	
0 -1	

## 2. Печери

**Максимальна оцінка: 100 балів**

**Обмеження на час: 7 сек.**

**Обмеження на пам'ять: 128 МБ**

**Вхідний файл: caves.in**

**Вихідний файл: caves.out**

**Програма: caves.\***

Пер Гюнт, герой п'єси норвезького письменника Генріка Ібсена, потрапив у полон короля тролів. Щоб визволитися, Пер Гюнт повинен перемогти у грі, яку король пропонує своїм бранцям. Правила гри такі:

- є 10 печер, деякі з яких сполучені між собою;
- бранець грає протягом  $t$  днів:
  - у перший день він може вибрати собі довільну печеру;
  - кожного наступного дня він може або залишитися у печері, у якій перебуває, або перейти у будь-яку іншу печеру, до якої є хід з поточної печери;
- кожного дня король винагороджує бранця певною наперед обумовленою кількістю золотих монет, яка залежить і від печери, і від номера дня;
- якщо за всю гру бранець накопичить кількість монет, виражену одним з  $l$  «королівських чисел», то бранця навіки залишають у полоні — він програв. Інакше його відпускають на волю з виграшем — усією накопиченою за час гри кількістю монет.

### **Завдання**

Визначити,  $N_w$ ,  $p_{\max}$ ,  $p_{\min}$  — відповідно кількість різних виграшів, найбільший і найменший виграші.

<b>Приклад</b>	
<b>caves.in</b>	<b>caves.out</b>
2 2	2 9 5
1 0 1 0 0 0 0 0 0 0	
0 1 0 0 0 0 0 0 0 0	
1 0 1 0 0 0 0 0 0 0	
0 0 0 1 0 0 0 0 0 0	
0 0 0 0 1 0 0 0 0 0	
0 0 0 0 0 1 0 0 0 0	
0 0 0 0 0 0 1 0 0 0	
0 0 0 0 0 0 0 1 0 0	
0 0 0 0 0 0 0 0 1 0	
0 0 0 0 0 0 0 0 0 1	
1 2 3 0 0 0 0 0 0 0	
4 5 6 0 0 0 0 0 0 0	
7 0	

### **Вхідні дані**

Перший рядок файлу містить десяткові записи натуральних чисел  $m, l$ , де  $m \leq 60$ .

Кожний з наступних 10 рядків містить по 10 цілих чисел — нулів або одиниць. При  $1 \leq j, k \leq 10$ : якщо  $k$  число  $(j + 1)$  рядка файлу дорівнює 1, то з  $j$  печери можна потрапити безпосередньо до  $k$  печери і навпаки. Інакше це зробити неможливо.

Кожний з наступних  $m$  рядків містить по 10 невід'ємних цілих чисел. При  $1 \leq j \leq m$  і  $1 \leq k \leq 10$ :  $k$  число  $(j + 11)$  рядка — кількість монет, які отримує гравець як винагороду за перебування у  $j$  день у  $k$  печері.

Останній  $(m + 12)$  рядок містить у порядку спадання  $l$  різних цілих невід'ємних «королівських чисел», кожне з яких менше від  $2^{64}$ . Останнє число у цьому рядку — нуль.

### **Вихідні дані**

Єдиний рядок файлу містить десяткові записи трьох чисел:  $N_w, p_{\max}, p_{\min}$ . Якщо  $N_w = 0$ , то  $p_{\max} = p_{\min} = 0$ . Відомо, що  $p_{\max} < 2^{64}$ ,  $l + N_w \leq 10^6$ .