

## 1. Chạy bộ (RUNNING.\*)

Mỗi buổi sáng có  $n$  người đi bộ trên một con đường dài vô hạn. Mỗi người xuất phát từ các vị trí khác nhau trên đường, tất cả đi về cùng một hướng, tuy nhiên vận tốc của mỗi người có thể khác nhau.

Mỗi khi một người đi bộ phía sau gặp một người đi bộ phía trước thì họ lập thành một nhóm đi bộ cùng nhau trò chuyện vui vẻ. Vận tốc của nhóm đi bộ này tất nhiên là bằng vận tốc của người đi bộ phía trước.

Tất cả  $n$  người đều đi bộ  $T$  phút. Hỏi rằng ở thời điểm kết thúc việc đi bộ thì có bao nhiêu nhóm người đi cùng nhau?

*Input:*

- Dòng đầu tiên chứa hai số nguyên dương  $n, T$  ( $1 \leq n \leq 10^5$ ;  $1 \leq T \leq 10^9$ )
- Tiếp theo là  $n$  dòng, mỗi dòng chứa hai số nguyên là vị trí xuất phát và vận tốc của một người. Vị trí xuất phát là số nguyên không âm còn vận tốc là số nguyên dương, cả hai có giá trị không quá  $10^9$ . Các vị trí xuất phát của những người khác nhau là khác nhau và trong file input chúng được liệt kê theo giá trị tăng dần.

*Output:* Một số nguyên duy nhất là số nhóm sau khi kết thúc buổi đi bộ.

*Example:*

| input                                  | output |
|--|--------|
| 5 3<br>0 1<br>1 2<br>2 3<br>3 2<br>6 1 | 3      |

Ghi chú:

- Subtask 1 (50%):  $n \leq 5000$
- Subtask 2 (50%):  $n \leq 10^5$

## 2. Bao nhiêu số (PERMN.\*)

Dũng là một học sinh giỏi toán, rất nổi tiếng ở trong trường. Thật không may là Mai lại không nghĩ như vậy. Để làm cho Mai thay đổi quan niệm đó về mình, Dũng nói rằng anh ta có thể giải quyết bất kỳ bài toán nào. Sau khi suy nghĩ, Mai nói với Dũng là hãy đếm xem có bao nhiêu số là đồng với  $n$  theo modulo  $m$ ?

Một số  $x$  được gọi là đồng với  $n$  theo modulo  $m$  nếu như:

- $x$  nhận được từ  $n$  bằng cách sắp xếp lại các chữ số của  $n$ .
- $x$  không có chữ số 0 ở đầu
- Phần dư của  $x$  khi chia cho  $m$  bằng 0.

Dũng tuy rất giỏi toán, nhưng số lượng số là quá nhiều đối với anh ta. Viết chương trình giúp anh ta giải quyết vấn đề này.

*Input:* Một dòng chứa hai số nguyên  $x$  ( $1 \leq x \leq 10^{18}$ ) và số  $m$  ( $1 \leq m \leq 100$ )

*Output:* Một số nguyên là kết quả tìm được

*Example:*

| Input     | Output |
|-----------|--------|
| 104 2     | 3      |
| 223 4     | 1      |
| 7067678 8 | 47     |

### 3. Xây cầu (BRIDGE.\*)

Byteland là một quần đảo có  $n$  hòn đảo. Các hòn đảo đánh số từ 1 đến  $n$ . Để đơn giản chúng ta hình dung mỗi hòn đảo như là một điểm trên mặt phẳng tọa độ. Giữa một số cặp đảo có những cây cầu nối trực tiếp. Để đơn giản có thể coi độ dài của những cây cầu này bằng khoảng cách giữa hai điểm mô tả hai đảo (chính xác hơn là nó tỷ lệ với khoảng cách này, tuy nhiên ta có thể bỏ qua hệ số tỷ lệ mà không làm thay đổi vấn đề).

Hỏi rằng có tồn tại một đường đi qua tất cả các đảo, mỗi đảo đúng một lần hay không?. Trong trường hợp không tồn tại đường đi như vậy hãy chỉ ra cách bổ sung thêm những cây cầu để thực hiện được điều đó với tổng độ dài những cây cầu cần xây dựng thêm là nhỏ nhất.

*Input:*

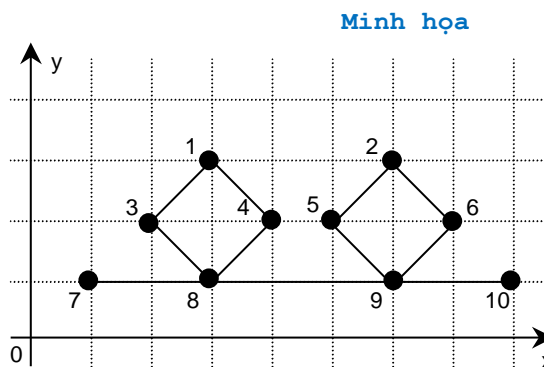
- Dòng 1: Ghi hai số nguyên dương  $n \leq 20$  - số lượng đảo và  $m$  là số lượng cây cầu hiện có
- $n$  dòng tiếp theo, mỗi dòng ghi hai số thực  $x_i, y_i$  là tọa độ của một hòn đảo
- $m$  dòng cuối cùng, mỗi dòng ghi hai số nguyên  $u, v$  ( $1 \leq u, v \leq n$ ) thể hiện một cây cầu hiện có nối đảo  $u$  với đảo  $v$

*Output:*

- Dòng 1 ghi "YES" (không dấu nháy kép) nếu như tồn tại một hành trình qua tất cả các đảo, mỗi đảo đúng một lần. Trường hợp ngược lại ghi "NO" (không dấu nháy kép)
- Nếu dòng 1 ghi "NO" thì dòng thứ hai ghi một số thực với 3 chữ số thập phân thể hiện tổng chiều dài nhỏ nhất của các cây cầu cần xây dựng thêm để tồn tại một hành trình qua tất cả các đảo, mỗi đảo đúng một lần.

*Example:*

| Input   | Output |
|---------|--------|
| 10 11   | NO     |
| 3.0 3.0 | 1.000  |
| 6.0 3.0 |        |
| 2.0 2.0 |        |
| 4.0 2.0 |        |
| 5.0 2.0 |        |
| 7.0 2.0 |        |
| 1.0 1.0 |        |
| 3.0 1.0 |        |
| 6.0 1.0 |        |
| 8.0 1.0 |        |
| 1 3     |        |
| 1 4     |        |
| 2 5     |        |
| 2 6     |        |
| 3 8     |        |
| 4 8     |        |
| 5 9     |        |
| 6 9     |        |
| 7 8     |        |
| 8 9     |        |
| 9 10    |        |



### 4. Tìm cửa thoát (EXIT.\*)

Một mê cung có dạng một cây nhị phân đầy đủ có độ sâu  $h$ . Các nút của cây được đánh số theo quy tắc:

- Gốc của cây được đánh số 1
- Với một nút trong  $i$  ( $i \leq 2^{h-1} - 1$ ) thì con trái của nó được đánh số là  $2i$  và con phải của nó được đánh số là  $2i + 1$

(độ sâu của một nút được định nghĩa bằng 1 nếu là nút gốc và bằng  $1 + \text{độ sâu của nút cha}$  với các nút khác. Với cách đánh số trên, các nút lá luôn ở độ sâu  $h$  và có số hiệu từ  $2^{h-1}$  đến  $2^h - 1$ )

Trong mê cung có duy nhất một nút lá có cửa thoát. Nhiệm vụ của bạn là phải tìm cửa thoát dựa theo thông tin trả về của  $q$  lần khảo sát: Thông tin mỗi lần khảo sát có dạng hoặc "1. Cửa thoát nằm trên lá với tổ tiên ở độ sâu  $i$  có số hiệu từ  $L$  đến  $R$ " hoặc "2. Cửa thoát không nằm trên các lá với tổ tiên ở độ sâu  $i$  có số hiệu từ  $L$  đến  $R$ "

Hỏi rằng với các thông tin trên có xác định được số hiệu nút lá có cửa thoát hay không?

**Input:**

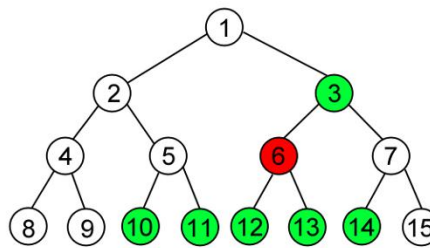
- Dòng đầu tiên ghi hai số nguyên dương  $h, q$  ( $1 \leq h \leq 50, 0 \leq q \leq 10^5$ )
- $q$  dòng tiếp theo, mỗi dòng ghi năm số  $i, L, R, ans$  ( $1 \leq i \leq h, 2^{i-1} \leq L \leq R \leq 2^i - 1, ans \in \{0, 1\}$ ) thể hiện thông tin của một khảo sát với  $ans = 1$  nếu thuộc dạng 1 và  $ans = 0$  nếu thuộc dạng 2

**Output:**

- Nếu dữ liệu khảo sát mâu thuẫn ghi "No" (không nháy kép)
- Nếu có nhiều lá thỏa mãn thì ghi "Multi" (không có nháy kép)
- Nếu có duy nhất một lá thỏa mãn ghi số hiệu của lá này

**Example:**

| input                                     | output |
|---|--------|
| 4 3<br>4 10 14<br>1<br>3 6 6 0<br>2 3 3 1 | 14     |



## 5. Các cặp số đẹp (BPN.\*)

Dãy các cặp số nguyên  $(a_1, b_1), (a_2, b_2), \dots, (a_k, b_k)$  được gọi là đẹp nếu như thỏa mã hai điều kiện dưới đây:

- $1 \leq a_1 \leq b_1 \leq a_2 \leq b_2 \leq \dots \leq a_k \leq b_k \leq n$  với  $n$  là một số cho trước
- Tất cả các số  $b_1 - a_1, b_2 - a_2, \dots, b_k - a_k$  là khác nhau

Cho số nguyên dương  $n$  hãy đếm số lượng dãy cặp đẹp độ dài  $k$ . Vì con số này có thể rất lớn nên bạn chỉ cần lấy phần dư của nó khi chia cho  $1000000007$  ( $10^9+7$ )

**Input:**

- Dòng đầu ghi số nguyên  $T$  ( $1 \leq T \leq 2 \cdot 10^5$ ) - số lượng bộ dữ liệu.
- $T$  dòng tiếp theo, mỗi dòng ghi hai số nguyên dương  $n, k$  ( $1 \leq k \leq n \leq 1000$ ) là một bộ dữ liệu.

**Output:** Với mỗi bộ dữ liệu, in ra kết quả tìm được tương ứng

**Example:**

| Input                                       | Output                     |
|---|----------------------------|
| 6<br>1 1<br>2 1<br>2 2<br>3 1<br>3 2<br>3 3 | 1<br>3<br>0<br>6<br>2<br>0 |