



## 誘拐 2 (Abduction 2)

ある晴天の日のこと、ある交差点で、誘拐事件が発生した。犯人は Anna と Bruno であり、二人は車に乗って誘拐現場から逃走したと考えられている。誘拐犯の車は未だ見つかっておらず、警察は今なお車の行方を追っている。

誘拐犯の車の走った市街地は、東西にのびる道路が  $H$  本、南北にのびる道路が  $W$  本走る碁盤目状の街である。2つの隣接する交差点間の距離は、1 キロメートルである。

それぞれの道路について、人通りの多さを表す整数が与えられる。北から  $i$  本目 ( $1 \leq i \leq H$ ) の東西方向の道路の人通りの多さは  $A_i$  であり、西から  $j$  本目 ( $1 \leq j \leq W$ ) の南北方向の道路の人通りの多さは  $B_j$  である。これら  $H+W$  個の値は互いに異なる。それぞれの道路について、道路の人通りの多さは、道路上のどの地点においても同じである。

また、警察による捜査の結果、誘拐犯は市街地で移動する際、次のような方法で移動をしていたことが明らかとなった。

- 市街地の外や、道路ではない場所に出ることはない。
- 最初は、誘拐犯は、誘拐した交差点から移動可能な方向のうちどれか 1 つを選んで移動する。
- ある交差点に差し掛かったとき、今まで移動してきた道路よりも交わっている道路の方が人通りが多いときは、その交差点で曲がる。もし、2 方向どちらにも曲がることのできる場合は、どちらを選んだ可能性もある。
- ある交差点に差し掛かったとき、今まで移動してきた道路の方が交わっている道路よりも人通りが多いときは、そのまま直進する。ただし、市街地の端におり直進できないときは、そこで移動を終了する。

誘拐現場の候補地として、 $Q$  箇所の交差点が候補に挙がっている。 $Q$  箇所の候補地は互いに異なる。警察は、調査チームの人数を決めるときの参考として、それぞれの誘拐現場の候補地に対し、誘拐事件がそこで発生したと仮定したとき、誘拐犯がその場所から最長で何キロメートル移動できるかを知りたい。

$Q$  個のクエリについて、与えられた誘拐現場の候補地から移動可能な経路の最長の長さを求めよ。

## 課題

市街地の道路の人通りの多さと、 $Q$  箇所の誘拐現場の候補地が与えられたとき、それぞれの候補地から移動可能な経路の長さの最大値を求めるプログラムを作成せよ。



## 入力

標準入力から以下の入力を読み込め。

- 1 行目には、3 個の整数  $H, W, Q$  が空白を区切りとして書かれている。これは、市街地は東西方向にのびる  $H$  本の道路と南北方向にのびる  $W$  本の道路からなり、誘拐現場の候補地として  $Q$  箇所が与えられることを表す。
- 2 行目には、 $H$  個の整数  $A_1, A_2, \dots, A_H$  が空白を区切りとして書かれている。これは、北から  $i$  本目 ( $1 \leq i \leq H$ ) の東西方向の道路の人通りの多さが  $A_i$  であることを表す。
- 3 行目には、 $W$  個の整数  $B_1, B_2, \dots, B_W$  が空白を区切りとして書かれている。これは、西から  $j$  本目 ( $1 \leq j \leq W$ ) の南北方向の道路の人通りの多さが  $B_j$  であることを表す。
- 続く  $Q$  行のうちの  $k$  行目 ( $1 \leq k \leq Q$ ) には、2 個の整数  $S_k, T_k$  が空白を区切りとして書かれている。これは、 $k$  番目の誘拐現場の候補地が、北から  $S_k$  本目の東西方向の道路と西から  $T_k$  本目の南北方向の道路の交差点であることを表す。

## 出力

標準出力に  $Q$  行出力せよ。  $k$  行目 ( $1 \leq k \leq Q$ ) には、 $k$  番目の誘拐現場の候補地から移動を開始したときの移動可能な経路の長さの最大値を、キロメートル単位で出力せよ。

## 制限

すべての入力データは以下の条件を満たす。

- $2 \leq H \leq 50\,000$ .
- $2 \leq W \leq 50\,000$ .
- $1 \leq Q \leq 100$ .
- $1 \leq A_i \leq 1\,000\,000\,000$  ( $1 \leq i \leq H$ ).
- $1 \leq B_j \leq 1\,000\,000\,000$  ( $1 \leq j \leq W$ ).
- $H + W$  個の整数  $A_1, A_2, \dots, A_H, B_1, B_2, \dots, B_W$  は互いに異なる。
- $1 \leq S_k \leq H$  ( $1 \leq k \leq Q$ ).
- $1 \leq T_k \leq W$  ( $1 \leq k \leq Q$ ).
- $(S_k, T_k) \neq (S_l, T_l)$  ( $1 \leq k < l \leq Q$ ).



## 小課題

この課題では小課題は全部で 5 個ある。各小課題の配点および追加の制限は以下の通りである。

### 小課題 1 [13 点]

- $H \leq 8$ .
- $W \leq 8$ .
- $Q = 1$ .

### 小課題 2 [10 点]

- $H \leq 2000$ .
- $W \leq 2000$ .
- $Q = 1$ .

### 小課題 3 [17 点]

- $Q = 1$ .

### 小課題 4 [4 点]

- $H \leq 2000$ .
- $W \leq 2000$ .

### 小課題 5 [56 点]

追加の制限はない。



## 入出力例

入力例 1	出力例 1
3 3 5	4
3 2 6	5
1 4 5	4
1 1	4
1 2	2
2 2	
3 1	
3 3	

例えば、3 番目のクエリに関しては、次のように移動する場合が最長となる。

- 北から 2 本目の道路と西から 2 本目の道路の交差点から、東方向に 1 キロメートル移動する。
- 北から 2 本目の道路と西から 3 本目の道路の交差点からは、南もしくは北に移動可能であり、南方向に 1 キロメートル移動する。
- 北から 3 本目の道路と西から 3 本目の道路の交差点からは、西にのみ移動可能であり、西方向に 1 キロメートル移動する。
- 北から 3 本目の道路と西から 2 本目の道路の交差点からは、西にのみ移動可能であり、西方向に 1 キロメートル移動する。
- 北から 3 本目の道路と西から 1 本目の道路の交差点からは、もうどの方向にも移動することはできないので、ここで移動を終了する。

このように移動すると、移動した経路の長さは 4 キロメートルとなる。

入力例 2	出力例 2
4 5 6	7
30 10 40 20	6
15 55 25 35 45	9
1 3	4
4 3	6
2 2	9
4 1	
2 5	
3 3	



## 都市 (City)

JOI 王国にはいくつかの都市があり，都市を結ぶ道路網は以下の条件を満たす．

- (条件 1) 都市には  $0, 1, \dots, N-1$  の番号が付けられている．ここで， $N$  は JOI 国の都市の個数である．
- (条件 2) 都市は  $N-1$  本の双方向に行き来可能な道路で結ばれており，すべての 2 つの異なる都市の間をいくつかの道路を通して行き来可能である．
- (条件 3) 都市 0 から他のどの都市へも，18 本以下の道路を通ることで行くことができる．

JOI 王国では，都市 0 からは毎日多くの国民が発し，他の都市へ向かう．多くの国民は，2 つの異なる都市を目的地としているため，次のような質問を行うことがある：2 つの異なる都市  $X, Y$  に対し，以下の (0), (1), (2) のどれが成り立つか？

- (0) 都市 0 から都市  $X$  へ向かう途中に，都市  $Y$  を必ず経由する．
- (1) 都市 0 から都市  $Y$  へ向かう途中に，都市  $X$  を必ず経由する．
- (2) (0), (1) のいずれでもない．

上の条件のもとで，(0), (1), (2) のうち，ちょうど一つが成り立つ．ただし， $X = 0$  のときは， $Y$  の値によらず (1) が成り立つとする．また， $Y = 0$  のときは， $X$  の値によらず (0) が成り立つとする．

他国においても，JOI 王国の道路網と同様に，(条件 1)～(条件 3) を満たす道路網が存在することが分かっている．そこで，他国でも使えるように，JOI 王国では，次のような 2 つの装置を開発することになった．

- (装置 1) 都市の個数  $N$  と道路網の情報を元に，それぞれの都市に対して符号を割り当てる．符号は， $0$  以上  $2^{60} - 1$  以下の整数値である．
- (装置 2) (装置 1) によって 2 つの異なる都市  $X, Y$  に割り当てられた符号を入力したとき，これらの都市に対して上記の質問に答える．

符号として大きな整数値が与えられると管理が大変であるため，できるだけ小さな符号を割り当てるような装置を開発したい．

(装置 2) を利用する際には，都市の個数  $N$  や道路網の情報は，入力として直接与えられないことに注意せよ．

## 課題

条件を満たす 2 つの装置を開発するために，次の 2 つのプログラムを作成せよ：

- 1 つ目のプログラムは，JOI 王国の都市の個数  $N$  と道路網の情報を受け取り，各都市に符号を割り当てる．
- 2 つ目のプログラムは，1 つ目のプログラムによって割り当てられた 2 つの異なる都市の符号を受け取り，これらの都市に対して質問に答える．



## 実装の詳細

あなたは同じプログラミング言語で2つのファイルを提出しなければならない。

1つ目のファイルは `Encoder.c` または `Encoder.cpp` という名前である。このファイルは装置1を実装したファイルであり、以下のルーチンを実装していなければならない。プログラムは `Encoder.h` をインクルードすること。

- `void Encode(int N, int A[], int B[])`

この関数は各テストケースにおいて1回だけ呼び出される。

- 引数  $N$  は都市の個数  $N$  を表す。
- 引数  $A[], B[]$  は長さ  $N-1$  の数列であり、道路網の情報を表す。要素  $A[i], B[i]$  ( $0 \leq i \leq N-2$ ) は、都市  $A[i]$  と都市  $B[i]$  の間が道路で直接結ばれていることを表す。

あなたのプログラムは以下の関数を呼び出さなければならない。

- ★ `void Code(int city, long long code)`

この関数は、都市に符号を割り当てる操作を表す。

- ◇ 引数 `city` は符号を割り当てる都市を表す。`city` は0以上  $N-1$  以下の整数値でなければならない。この範囲外の値を指定して呼び出した場合、不正解 [1] と判定される。また、同じ引数 `city` で2回以上呼び出すことはできない。同じ引数で2回呼び出した場合、不正解 [2] と判定される。
- ◇ 引数 `code` は都市 `city` に割り当てる符号を表す。`code` は0以上  $2^{60}-1$  以下の整数値でなければならない。この範囲外の値を指定して呼び出した場合、不正解 [3] と判定される。

あなたのプログラムは、関数 `Code` をちょうど  $N$  回呼び出さなければならない。関数 `Encode` の実行の終了時に関数 `Code` の呼び出し回数が  $N$  でなかった場合、不正解 [4] と判定される。

関数 `Encode` の呼び出しが不正解と判定された場合、その時点でプログラムは終了する。

2つ目のファイルは `Device.c` または `Device.cpp` という名前である。このファイルは装置2を実装したファイルであり、以下のルーチンを実装していなければならない。プログラムは `Device.h` をインクルードすること。

- `void InitDevice()`

この関数は装置2の初期化に対応する。次に示す関数 `Answer` が呼び出される前に、`InitDevice` が1回だけ呼び出される。

- `int Answer(long long S, long long T)`

この関数は各質問に対応する。各質問に対して `Answer` が1回ずつ呼び出される。

- 引数  $S, T$  は、2つの異なる都市  $X, Y$  に関数 `Encode` の中で割り当てられた符号である。



- 関数 `Answer` は、質問の答えとして、次の条件に従った値を返さなければならない。
  - ◇ 都市 0 から都市  $X$  へ向かう途中に、都市  $Y$  を必ず経由する場合は、0 を返すこと。
  - ◇ 都市 0 から都市  $Y$  へ向かう途中に、都市  $X$  を必ず経由する場合は、1 を返すこと。
  - ◇ 上のいずれの条件も満たさない場合は、2 を返すこと。

すなわち、`Answer` は 0 以上 2 以下の整数値を返すべきである。この範囲外の値を返した場合は、不正解 [5] と判定される。また、値が範囲内であっても、これらの条件に従わない値を返した場合は、不正解 [6] と判定される。

## 採点の手順

採点は以下の手順で行われる。不正解と判定された場合はその時点でプログラムは終了される。

- (1) 関数 `Encode` が 1 回呼び出される。
- (2) 関数 `InitDevice` が 1 回呼び出される。
- (3) テストケースにおいて、装置 2 に与えられる質問が  $Q$  個ある。 $j$  番目 ( $1 \leq j \leq Q$ ) の質問では、関数 `Encode` の中で都市  $X_j, Y_j$  に割り当てられた符号をそれぞれ  $S_j, T_j$  とし、関数 `Answer` を、引数  $S$  を  $S_j$ 、引数  $T$  を  $T_j$  として呼び出す。
- (4) 正解となる。

## 重要な注意

- 実行時間計測・使用メモリ計測の対象となるのは、「採点の手順」における手順 (1), (2), (3) である。手順 (3) において `Answer` は合計  $Q$  回呼び出されることに注意せよ。
- 手順 (1) における `Encode` の呼び出し、手順 (2) における `InitDevice` の呼び出しおよび、手順 (3) におけるすべての `Answer` の呼び出しにおいて、不正解と判定されたり、実行時エラーとなってはならない。
- 内部での使用のために他のルーチンを実装したり、グローバル変数を宣言するのは自由である。ただし、提出された 2 つのプログラムは、採点プログラムとまとめてリンクされて 1 つの実行ファイルになるので、各ファイル内のすべてのグローバル変数と内部ルーチンを `static` で宣言して、他のファイルとの干渉を避ける必要がある。採点時には、このプログラムは装置 1 側、装置 2 側として 2 個のプロセスとして実行されるので、装置 1 側と装置 2 側でプログラム中のグローバル変数を共有することはできない。
- あなたの提出は標準入力・標準出力、あるいは他のファイルといかなる方法でもやりとりしてはならない。



## コンパイル・実行の方法

作成したプログラムをテストするための、採点プログラムのサンプルが、コンテストサイトからダウンロードできるアーカイブの中に含まれている。このアーカイブには、提出しなければならないファイルのサンプルも含まれている。

採点プログラムのサンプルは1つのファイルからなる。そのファイルは `grader.c` または `grader.cpp` である。作成したプログラムを `Encoder.c` および `Device.c`, または `Encoder.cpp` および `Device.cpp` とするとき、作成したプログラムをテストするには、次のようにコマンドを実行する。

- C の場合

```
gcc -std=c11 -O2 -o grader grader.c Encoder.c Device.c -lm
```

- C++ の場合

```
g++ -std=c++14 -O2 -o grader grader.cpp Encoder.cpp Device.cpp
```

コンパイルが成功すれば、`grader` という実行ファイルが生成される。

実際の採点プログラムは、採点プログラムのサンプルとは異なることに注意すること。採点プログラムのサンプルは単一のプロセスとして起動する。このプログラムは、標準入力から入力を読み込み、標準出力に結果を出力する。

## 採点プログラムのサンプルの入力

採点プログラムのサンプルは標準入力から以下の入力を読み込む。

- 1行目には、2個の整数  $N, Q$  が空白を区切りとして書かれている。これは都市が  $N$  個あり、与えられる質問が  $Q$  個あることを表す。
- 続く  $N-1$  行のうちの  $i+1$  行目 ( $0 \leq i \leq N-2$ ) には、2個の整数  $A_i$  と  $B_i$  が空白を区切りとして書かれている。これは都市  $A_i$  と都市  $B_i$  の間が道路で直接結ばれていることを表す。
- 続く  $Q$  行のうちの  $j$  行目 ( $1 \leq j \leq Q$ ) には、3個の整数  $X_j, Y_j, E_j$  が空白を区切りとして書かれている。これは  $j$  番目の質問において、 $X = X_j$  かつ  $Y = Y_j$  であり、あなたのプログラムがこの質問に対して  $E_j$  以外の解答を返したときに採点プログラムのサンプルが不正解と判定することを表す。

## 採点プログラムのサンプルの出力

採点プログラムのサンプルは標準出力へ以下の情報を出力する（引用符は実際には出力されない）。

- 正解の場合、都市に割り当てられた符号の整数の最大値が “Accepted : max\_code=123456” のように出力される。
- 不正解の場合、不正解の種類が “Wrong Answer [1]” のように出力される。

実行するプログラムが複数の不正解の条件を満たした場合、表示される不正解の種類はそれらのうち1つのみである。



## 制限

すべての入力データは以下の条件を満たす。  $N, Q, A_i, B_i, X_j, Y_j$  の意味については「採点プログラムのサンプルの入力」の項目を参照せよ。

- $2 \leq N \leq 250\,000$ .
- $1 \leq Q \leq 250\,000$ .
- $0 \leq A_i \leq N - 1$  ( $0 \leq i \leq N - 2$ ).
- $0 \leq B_i \leq N - 1$  ( $0 \leq i \leq N - 2$ ).
- $A_i \neq B_i$  ( $0 \leq i \leq N - 2$ ).
- 都市 0 から他のどの都市までも、18 本以下の道路を通ることで行くことができる。
- $0 \leq X_j \leq N - 1$  ( $1 \leq j \leq Q$ ).
- $0 \leq Y_j \leq N - 1$  ( $1 \leq j \leq Q$ ).
- $X_j \neq Y_j$  ( $1 \leq j \leq Q$ ).

## 小課題

この課題では小課題は全部で 2 個ある。各小課題の配点および追加の制限は以下の通りである。

### 小課題 1 [8 点]

- $N \leq 10$ .

### 小課題 2 [92 点]

追加の制限はない。この小課題では、以下に従い得点が決定される。

- この小課題のすべてのテストケースにおける、都市に割り当てられた符号の整数の最大値を  $L$  とおく。
- このとき、この小課題の得点は、
  - $2^{38} \leq L$  のとき、0 点。
  - $2^{36} \leq L \leq 2^{38} - 1$  のとき、10 点。
  - $2^{35} \leq L \leq 2^{36} - 1$  のとき、14 点。
  - $2^{34} \leq L \leq 2^{35} - 1$  のとき、22 点。
  - $2^{28} \leq L \leq 2^{34} - 1$  のとき、 $\lfloor 372 - 10 \log_2(L + 1) \rfloor$  点 ( $\lfloor x \rfloor$  は  $x$  を超えない最大の整数)。
  - $L \leq 2^{28} - 1$  のとき、92 点。

競技システムにおいては、 $2^{38} \leq L$  のときは、この小課題の詳細が「正解：0 点」のように表示されることも、「不正解」と表示されることもあることに注意せよ。



## やり取りの例

採点プログラムのサンプルが読み込む入力の例と、それに対応する関数の呼び出しの例を以下に示す。

入力例	ルーチンの呼び出しの例	
6 5	装置 1	装置 2
4 1	Encode(...)	
0 3	Code(0,0)	
4 5	Code(2,4)	
3 2	Code(4,16)	
3 4	Code(1,1)	
2 4 2	Code(3,9)	
1 0 0	Code(5,25)	
5 1 2		InitDevice()
5 3 0		Answer(4,16)
4 1 1		Answer(1,0)
		Answer(25,1)
		Answer(25,9)
		Answer(16,1)

このとき、Encode(...) に渡される引数は次の通りである。

引数	Encode(...)
N	6
A	{4,0,4,3,3}
B	{1,3,5,2,4}



## ドラゴン 2 (Dragon 2)

JOI 平原では、人々はドラゴンたちと共に生きている。

JOI 平原は広大な座標平面であり、直交する X 軸と Y 軸が設定されている。X 座標が  $x$ , Y 座標が  $y$  の点を  $(x, y)$  で表す。

JOI 平原には  $N$  匹のドラゴンが生息しており、1 から  $N$  の番号が付けられている。また、ドラゴンには  $M$  種の種族があり、1 から  $M$  の番号が付けられている。ドラゴン  $i$  ( $1 \leq i \leq N$ ) は常に JOI 平原の点  $(A_i, B_i)$  におり、その種族は  $C_i$  である。JOI 平原には、すべての種族のドラゴンが生息しているとは限らない。

JOI 平原には人間の住む村が 2 つあり、それぞれ JOI 平原の点  $(D_1, E_1)$  および点  $(D_2, E_2)$  に位置する。2 つの村は道で結ばれており、その道は村が存在する 2 点を結ぶ線分である。

点  $(A_1, B_1), \dots, (A_N, B_N)$  および点  $(D_1, E_1), (D_2, E_2)$  は、互いに異なり、これらの内のどの 3 点も同一直線上にない。

ドラゴンたちはしばしば種族間で対立する。種族  $a$  ( $1 \leq a \leq M$ ) が種族  $b$  ( $1 \leq b \leq M, a \neq b$ ) に対して敵意を抱くと、種族  $a$  に属するそれぞれのドラゴンが、種族  $b$  に属するすべてのドラゴンに向けて火の玉を吐く。火の玉は標的のドラゴンに向かって直進し、標的に到達したあとも同じ方向に飛び去っていく。すなわち、火の玉の迎撃経路は半直線である。

種族間対立の際に、ドラゴンの吐いた火の玉が道を横切れば、道はダメージを受けるだろう。あなたは近い将来に起こりうるドラゴンたちの種族間対立として、考えられる  $Q$  個の可能性をリストアップした。想定される種族間対立のそれぞれについて、道を横切る火の玉の個数を求めたい。

### 課題

ドラゴンたちおよび人間の村の情報と、ドラゴンたちの種族間対立の可能性のリストが与えられるとき、それぞれの種族間対立について、道を横切る火の玉の個数を求めるプログラムを作成せよ。

### 入力

標準入力から以下のデータを読み込め。

- 1 行目には、2 個の整数  $N, M$  が空白を区切りとして書かれている。これは、JOI 平原にはドラゴンが  $N$  匹おり、 $M$  種の種族に分かれていることを表す。
- 続く  $N$  行のうちの  $i$  行目 ( $1 \leq i \leq N$ ) には、3 個の整数  $A_i, B_i, C_i$  が空白を区切りとして書かれている。これは、ドラゴン  $i$  が JOI 平原の点  $(A_i, B_i)$  におり、種族が  $C_i$  であることを表す。
- 次の行には、4 個の整数  $D_1, E_1, D_2, E_2$  が空白を区切りとして書かれている。これは、人間の住む村が JOI 平原の点  $(D_1, E_1)$  および点  $(D_2, E_2)$  に位置していることを表す。
- 次の行には、整数  $Q$  が書かれている。これは、想定される種族間対立が  $Q$  個あることを表す。



- 続く  $Q$  行のうちの  $j$  行目 ( $1 \leq j \leq Q$ ) には, 2 個の整数  $F_j, G_j$  が空白を区切りとして書かれている. これは, 想定される種族間対立の  $j$  個目において, 種族  $F_j$  が種族  $G_j$  に対して敵意を抱くことを表す.

## 出力

出力は  $Q$  行からなる. 標準出力の  $j$  行目 ( $1 \leq j \leq Q$ ) には,  $j$  個目の種族間対立において道を横切る火の玉の個数を出力せよ.

## 制限

すべての入力データは以下の条件を満たす.

- $2 \leq N \leq 30\,000$ .
- $2 \leq M \leq N$ .
- $-1\,000\,000\,000 \leq A_i \leq 1\,000\,000\,000$  ( $1 \leq i \leq N$ ).
- $-1\,000\,000\,000 \leq B_i \leq 1\,000\,000\,000$  ( $1 \leq i \leq N$ ).
- $1 \leq C_i \leq M$  ( $1 \leq i \leq N$ ).
- $-1\,000\,000\,000 \leq D_1 \leq 1\,000\,000\,000$ .
- $-1\,000\,000\,000 \leq E_1 \leq 1\,000\,000\,000$ .
- $-1\,000\,000\,000 \leq D_2 \leq 1\,000\,000\,000$ .
- $-1\,000\,000\,000 \leq E_2 \leq 1\,000\,000\,000$ .
- $N + 2$  個の点  $(A_1, B_1), \dots, (A_N, B_N), (D_1, E_1), (D_2, E_2)$  は互いに異なり, どの 3 点も同一直線上にない.
- $1 \leq Q \leq 100\,000$ .
- $1 \leq F_j \leq M$  ( $1 \leq j \leq Q$ ).
- $1 \leq G_j \leq M$  ( $1 \leq j \leq Q$ ).
- $F_j \neq G_j$  ( $1 \leq j \leq Q$ ).
- $(F_j, G_j) \neq (F_k, G_k)$  ( $1 \leq j < k \leq Q$ ).



## 小課題

この課題では小課題は全部で3個ある。各小課題の配点および追加の制限は以下の通りである。

### 小課題 1 [15 点]

- $N \leq 3000$ .

### 小課題 2 [45 点]

- $Q \leq 100$ .

### 小課題 3 [40 点]

追加の制限はない。

## 入出力例

入力例 1	出力例 1
4 2	1
0 1 1	2
0 -1 1	
1 2 2	
-6 1 2	
-2 0 2 0	
2	
1 2	
2 1	

1 個目の種族間対立では、

- ドラゴン 1 がドラゴン 3 に向けて吐いた火の玉は道を横切らない。
- ドラゴン 1 がドラゴン 4 に向けて吐いた火の玉は道を横切らない。
- ドラゴン 2 がドラゴン 3 に向けて吐いた火の玉は道を横切る。
- ドラゴン 2 がドラゴン 4 に向けて吐いた火の玉は道を横切らない。

よって道を横切る火の玉は 1 個である。

2 個目の種族間対立では、

- ドラゴン 3 がドラゴン 1 に向けて吐いた火の玉は道を横切る。



- ドラゴン 3 がドラゴン 2 に向けて吐いた火の玉は道を横切る.
- ドラゴン 4 がドラゴン 1 に向けて吐いた火の玉は道を横切らない.
- ドラゴン 4 がドラゴン 2 に向けて吐いた火の玉は道を横切らない.

よって道を横切る火の玉は 2 個である.

入力例 2	出力例 2
3 2 -10000000000 -1 1 -9999999998 -1 1 0 0 2 9999999997 1 999999999 1 1 1 2	1

入力例 3	出力例 3
6 3 2 -1 1 1 0 1 0 3 2 2 4 2 5 4 3 3 9 3 0 0 3 3 6 1 2 1 3 2 1 2 3 3 1 3 2	4 2 4 0 2 1