

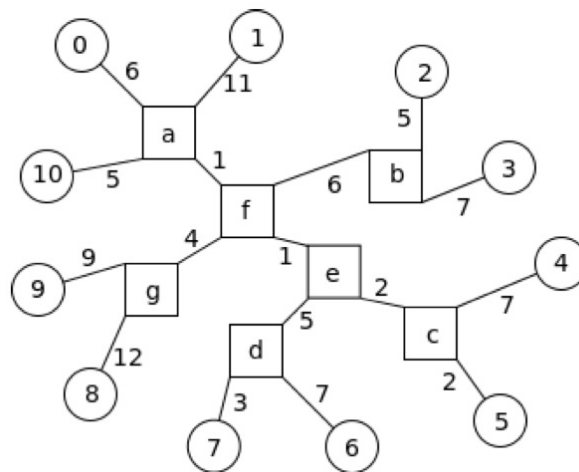
## 町 (Towns)

カザフスタンには  $N$  個の小さな町があり、それぞれ  $0$  から  $N - 1$  までの番号がつけられている。また、個数は不明だが、いくつかの大きな都市がある。カザフスタンの町、都市はいずれも居住地と呼ばれている。

カザフスタンのすべての居住地は、双方向に行き来可能な高速道路からなる  $1$  つのネットワークで結ばれている。それぞれの高速道路は  $2$  つの異なる居住地を結んでおり、どの  $2$  つの居住地についても、それらを直接結ぶような高速道路は最大でも  $1$  個しか存在しない。どの  $2$  つの居住地  $a, b$  に対しても、同じ高速道路を  $1$  回までしか使わないような  $a$  から  $b$  への経路がちょうど  $1$  個存在する。

それぞれの町は、ちょうど  $1$  個の他の居住地とのみ直接結ばれており、それぞれの都市は、 $3$  個以上の他の居住地と直接結ばれていることがわかっている。

下に示す図は、 $11$  個の町、 $7$  個の都市からなるネットワークを表している。この図において、町は丸で示され、数字により区別されている。また都市は四角で示され、英字で区別されている。



すべての高速道路には、正の整数値である長さが定まっている。 $2$  つの居住地の間の距離は、片方の居住地から他方の居住地まで移動するときに通る高速道路の長さの和の最小値として定められる。

すべての都市  $C$  に対して、距離  $r(C)$  を、 $C$  から最も遠い町までの距離として定める。都市  $C$  は、すべての都市の中で最小の距離  $r(C)$  を持つとき、ハブと呼ばれる。ハブと、そこから最も遠い町までの距離は  $R$  で表される。すなわち、 $R$  は  $r(C)$  の最小値である。

上の例において、都市  $a$  から最も遠い町は町  $8$  であり、それらの間の距離は  $r(a) = 1 + 4 + 12 = 17$  である。都市  $g$  に対して、同様に  $r(g) = 17$  である (都市  $g$

から最も遠い町の  $1$  つは町  $6$  である). 上の例における唯一のハブは都市  $f$  で,  $r(f) = 16$  である. よって上の例において  $R = 16$  である.

ハブをネットワークから取り除くと, ネットワークはいくつかの連結な部分に分割される. ハブは, 分割されたそれぞれの部分が最大でも  $\lfloor N/2 \rfloor$  個の町しか含んでいないとき, 偏りが無いと呼ばれる (このとき, 都市の数は数えないことに注意せよ). ここで,  $\lfloor x \rfloor$  は  $x$  を超えない最大の整数を表す.

先ほどの例において, 都市  $f$  はハブである. もし都市  $f$  を取り除いたなら, ネットワークは  $4$  個の部分に分割される. 分割される各部分に含まれる町はそれぞれ,  $\{0, 1, 10\}$ ,  $\{2, 3\}$ ,  $\{4, 5, 6, 7\}$ ,  $\{8, 9\}$  である.  $\lfloor 11/2 \rfloor = 5$  個より多くの町を含むような部分は存在しないため,  $f$  は偏りが無いハブである.

## 課題 (Task)

最初, 居住地と高速道路のネットワークについてあなたに与えられる情報は, 町の数  $N$  のみである. あなたは都市の数を知らないし, カザフスタンの高速道路ネットワークの構造に関することも何も知らない. あなたは,  $2$  つの町間の距離を問う質問をすることによってのみ, 新たな情報を得ることができる.

あなたの課題は, 次を決定することである:

- すべての小課題において, 距離  $R$ .
- 小課題  $3$  から  $6$  においては, ネットワーク中に偏りが無いハブが存在するかどうか.

あなたは, 関数 `hubDistance` を実装しなければならない. 採点プログラムは,  $1$  回の実行において, 複数のテストケースに対してあなたのプログラムを実行する.  $1$  回の実行におけるテストケースの数は  $40$  個以下である. それぞれのテストケースにおいて, 採点プログラムはあなたが実装した関数 `hubDistance` をちょうど  $1$  回だけ呼び出す. この関数は, 呼ばれるたびに必要な変数を必ず初期化するようにせよ.

- `hubDistance(N, sub)`
  - $N$ : 町の数である.
  - `sub`: 小課題の番号である (問題文では, 小課題の節において説明される).
  - `sub` が  $1, 2$  のいずれかである時には, この関数は  $R$  または  $-R$  のいずれかを返さなければならない.
  - `sub` が  $2$  より大きい時には, この関数は, もし偏りが無いハブが存在するなら  $R$  を, さもなくば  $-R$  を返さなければならない.

関数 `hubDistance` は, 関数 `getDistance(i, j)` を呼び出すことによって, 高速道路のネットワークに関する情報を得ることができる. この関数は, 町  $i, j$  の間の距離を返す. もし  $i, j$  が等しいならば, この関数は  $0$  を返す. ただし, 引数が無効な場合にもこの関数は  $0$  を返す.

## 小課題 (Subtasks)

それぞれのテストケースに対して,

- $N$  は 6 以上 110 以下である.
- 2 つの異なる町の間距離は 1 以上 1,000,000 以下である.

あなたのプログラムが行うことのできる質問の回数は制限されている。この制限は小課題ごとに異なり、下の表に示されている。もしあなたのプログラムが制限回数を超えて質問を呼びだそうとした場合には、プログラムは終了され、間違った答えを返したものとして扱われる。

小課題	点数	質問の回数	偏りがないハブの発見	追加の制約
1	13	$\frac{N(N-1)}{2}$	NO	なし
2	12	$\lceil \frac{7N}{2} \rceil$	NO	なし
3	13	$\frac{N(N-1)}{2}$	YES	なし
4	10	$\lceil \frac{7N}{2} \rceil$	YES	各都市は、ちょうど 3 つの居住地と結ばれている。
5	13	$5N$	YES	なし
6	39	$\lceil \frac{7N}{2} \rceil$	YES	なし

ここで、 $\lceil x \rceil$  は  $x$  以上の最小の整数を表す。

### 採点用プログラムのサンプル (Sample grader)

小課題の番号も入力の一部であることに注意せよ。採点用プログラムのサンプルの振る舞いは、小課題の番号によって変わる。

採点用プログラムのサンプルは、次のフォーマットに従って towns.in から入力を読み込む：

- 1 行目：小課題の番号および、テストケースの数。
- 2 行目：最初のテストケースにおける町の数  $N_1$ 。
- 続く  $N_1$  行：これらの行のうち  $i$  行目 ( $1 \leq i \leq N_1$ ) の  $j$  番目 ( $1 \leq j \leq N_1$ ) の数は、町  $i-1$  と町  $j-1$  の間の距離である。
- 以降のテストケースが続く。これらも最初のテストケースと同様のフォーマットで与えられる。

それぞれのテストケースにおいて、採点用プログラムのサンプルは hubDistance を呼び出し、その戻り値および、行われた質問の回数を別々の行に出力する。

先ほどの例に対応する入力ファイルは次の通りである：

```
1 1
11
0 17 18 20 17 12 20 16 23 20 11
17 0 23 25 22 17 25 21 28 25 16
18 23 0 12 21 16 24 20 27 24 17
20 25 12 0 23 18 26 22 29 26 19
17 22 21 23 0 9 21 17 26 23 16
12 17 16 18 9 0 16 12 21 18 11
20 25 24 26 21 16 0 10 29 26 19
16 21 20 22 17 12 10 0 25 22 15
23 28 27 29 26 21 29 25 0 21 22
20 25 24 26 23 18 26 22 21 0 19
11 16 17 19 16 11 19 15 22 19 0
```

このフォーマットは、高速道路のリストを直接指定するものとは大きく異なる。  
あなたは、異なる入力フォーマットを用いるために、採点用プログラムのサンプルを改変してもかまわない。