World directors (`directors`)

Linus and Tinus have recently been appointed as the new two *Directors of the World*.

The world is formed by N cities, numbered from 0 to $N-1$, and $N-1$ two-way roads. It is possible to move between any pair of cities using some sequence of roads. The distance between two cities is the minimum number of roads that needs to be traversed to move from one city to the other.

As newly appointed directors, Linus and Tinus are expected to perform a traditional *patrol* of the whole world, which consists of the following:

- First, Linus and Tinus move to some initial cities X and Y.
- Then, each day one of the two directors moves from their current city to one reachable with a single road from it. This continues until each of the directors have been in each city at least once and they have returned to their initial cities. Note that a director can move for multiple days in a row: the two directors do not need to alternate.

Linus and Tinus know very well that their patrol will be considered more solemn the more distant they are from each other: the *solemnity* of a patrol is the distance between the two directors when they are closest to each other.

The two directors have therefore hired you to help them plan their patrol and your job is to answer Q questions of the following type:

- If Linus starts on city X and Tinus starts on city Y, what is the maximum solemnity they can achieve with their patrol?

Implementation

You must submit a single file with the extension `.cpp`.

⇦ Among the attachments for this task, you will find a template `directors.cpp` with an example implementation.

You must implement the following functions:

C++
```
void init(int N, vector<int> A, vector<int> B);
```

- The integer N is the number of cities.
- The arrays A and B, indexed from 0 to $N-2$, contain the roads. In particular, the i-th road connects A_i and B_i.
- The function will be called once at the beginning of your program execution.

C++
```
int patrol(int X, int Y);
```

- The integers X and Y are the starting cities of Linus and Tinus.
- The function should return the maximum solemnity of a patrol where Linus and Tinus start at cities X and Y.
- The function will be called Q times during the execution of your program.

Sample Grader

A simplified version of the grader used during the correction is available in the directory related to this problem. You can use it to test your solutions locally. The sample grader reads input data from `stdin`, calls the function you need to implement, and writes to `stdout` in the following format.

The input file is made up of $N + Q$ lines, containing:

- Line 1: the integers N and Q.
- Line $2 + i$ ($0 \leq i < N - 1$): the integers A_i and B_i.
- Line $N + 1 + j$ ($0 \leq j < Q$): the integers X_j and Y_j.

The output file is made of Q lines, containing the values returned by the function `patrol`.

Constraints

- $1 \leq N \leq 200\,000$.
- $1 \leq Q \leq 100\,000$.
- $0 \leq X, Y < N$ in each query.

Scoring

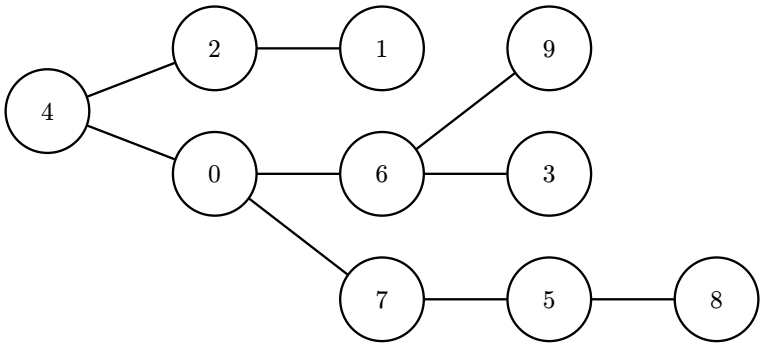
- **Subtask 0 [0 points]**: Example.
- **Subtask 1 [8 points]**: $A_i = 0$, $B_i = i + 1$ for all $0 \leq i < N - 1$.
- **Subtask 2 [16 points]**: $A_i = 0$, $B_i = i + 1$ or $A_i = i$, $B_i = i + 1$ for all $0 \leq i < N - 1$.
- **Subtask 3 [13 points]**: $N, Q \leq 200$.
- **Subtask 4 [14 points]**: $N \leq 1000$.
- **Subtask 5 [18 points]**: In each query, X and Y maximize the answer over all starting cities.
- **Subtask 6 [17 points]**: $Q \leq 200$.
- **Subtask 7 [14 points]**: No additional constraints.

Examples

stdin	stdout
10 3 0 4 1 2 8 5 6 0 9 6 2 4 7 0 3 6 5 7 9 8 0 6 6 4	2 1 2

Explanation

In **the first sample case** the world has the following structure:



For the first query we can achieve a patrol of solemnity 2 with the following moves:

- Linus takes the path: $9 \to 6 \to 3 \to 6 \to 0 \to 4 \to 2 \to 1$;
- Tinus takes the path: $8 \to 5 \to 7 \to 0 \to 6 \to 3 \to 6 \to 9$;
- Linus takes the path: $1 \to 2 \to 4 \to 0 \to 7 \to 5 \to 8$;
- Tinus takes the path: $9 \to 6 \to 3 \to 6 \to 0 \to 4 \to 2 \to 1$;
- Linus takes the path: $8 \to 5 \to 7 \to 0 \to 6 \to 3 \to 6 \to 9$;
- Tinus takes the path: $1 \to 2 \to 4 \to 0 \to 7 \to 5 \to 8$.

It can be shown that no patrol of solemnity 3 or more exists.