

World directors (directors)

Lino e Tino sono stati recentemente nominati come i due nuovi *Direttori al Mondo*.

Il mondo è formato da N città, numerate da 0 a $N - 1$, e $N - 1$ strade a doppio senso. È possibile spostarsi tra qualsiasi coppia di città usando una sequenza di strade. La distanza tra due città è il numero minimo di strade che devono essere percorse per spostarsi da una città all'altra.

In qualità di direttori appena nominati, Lino e Tino devono effettuare una tradizionale *pattuglia* di tutto il mondo, che consiste in quanto segue:

- Prima, Lino e Tino si spostano in alcune città iniziali X e Y .
- Poi, ogni giorno, uno dei due direttori si sposta dalla sua città attuale a una città adiacente. Si continua così finché entrambi i direttori non hanno visitato ogni città almeno una volta e non sono tornati alle loro città di partenza. Nota che un direttore può muoversi per più giorni di fila: i due non devono per forza alternarsi.

Lino e Tino sanno molto bene che la loro pattuglia sarà considerata più solenne quanto più distanti saranno l'uno dall'altro: la *solennità* di una pattuglia è la distanza tra i due direttori nel momento in cui sono più vicini.

I due direttori ti hanno quindi assunto per aiutarli a pianificare la loro pattuglia e il tuo compito è rispondere a Q domande del seguente tipo:

- Se Lino parte dalla città X e Tino parte dalla città Y , qual è la massima solennità che possono ottenere con la loro pattuglia?

Implementazione

Devi inviare un singolo file con estensione `.cpp`.



Tra gli allegati di questo problema, troverai un template `directors.cpp` con un'implementazione d'esempio.

Devi implementare le seguenti funzioni:

C++

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

- L'intero N è il numero di città.
- Gli array A e B , indicizzati da 0 a $N - 2$, contengono le strade. In particolare, la i -esima strada collega A_i e B_i .
- La funzione sarà chiamata una sola volta all'inizio dell'esecuzione del tuo programma.

C++

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

- Gli interi X e Y sono le città di partenza di Lino e Tino.
- La funzione dovrebbe restituire la massima solennità di una pattuglia in cui Lino e Tino partono dalle città X e Y .
- La funzione sarà chiamata Q volte durante l'esecuzione del tuo programma.

Grader di prova

Una versione semplificata del grader usato durante la correzione è disponibile nella cartella relativa a questo problema. Puoi usarla per testare le tue soluzioni in locale. Il grader d'esempio legge i dati di input da `stdin`, chiama le funzioni che devi implementare e scrive su `stdout` nel seguente formato.

Il file di input è composto da $N + Q$ linee, contenenti:

- Linea 1: gli interi N e Q .
- Linea $2 + i$ ($0 \leq i < N - 1$): gli interi A_i e B_i .
- Linea $N + 1 + j$ ($0 \leq j < Q$): gli interi X_j e Y_j .

Il file di output è composto da Q linee, contenenti i valori restituiti dalla funzione `patrol`.

Assunzioni

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

Assegnazione del punteggio

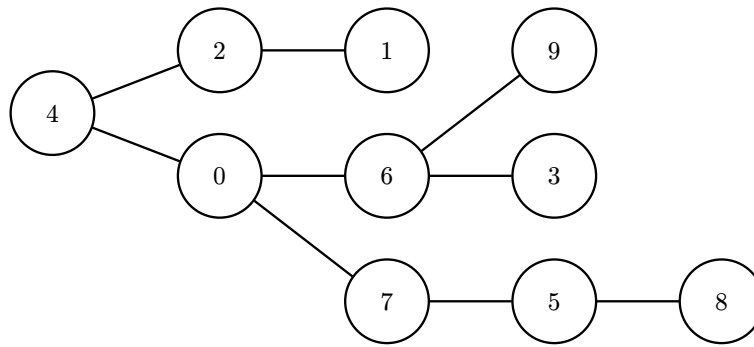
- **Subtask 0** [0 punti]: Caso d'esempio.
- **Subtask 1** [8 punti]: $A_i = 0$, $B_i = i + 1$ per ogni $0 \leq i < N - 1$.
- **Subtask 2** [16 punti]: $A_i = 0$, $B_i = i + 1$ o $A_i = i$, $B_i = i + 1$ per ogni $0 \leq i < N - 1$.
- **Subtask 3** [13 punti]: $N, Q \leq 200$.
- **Subtask 4** [14 punti]: $N \leq 1000$.
- **Subtask 5** [18 punti]: In ogni query, X e Y massimizzano la risposta su tutte le possibili città di partenza.
- **Subtask 6** [17 punti]: $Q \leq 200$.
- **Subtask 7** [14 punti]: Nessuna limitazione aggiuntiva.

Esempi di input/output

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

Spiegazione

Nel **primo caso d'esempio** il mondo ha la seguente struttura:



Per la prima query possiamo ottenere una pattuglia di solennità 2 con le seguenti mosse:

- Lino segue il percorso: $9 \rightarrow 6 \rightarrow 3 \rightarrow 6 \rightarrow 0 \rightarrow 4 \rightarrow 2 \rightarrow 1$;
- Tino segue il percorso: $8 \rightarrow 5 \rightarrow 7 \rightarrow 0 \rightarrow 6 \rightarrow 3 \rightarrow 6 \rightarrow 9$;
- Lino segue il percorso: $1 \rightarrow 2 \rightarrow 4 \rightarrow 0 \rightarrow 7 \rightarrow 5 \rightarrow 8$;
- Tino segue il percorso: $9 \rightarrow 6 \rightarrow 3 \rightarrow 6 \rightarrow 0 \rightarrow 4 \rightarrow 2 \rightarrow 1$;
- Lino segue il percorso: $8 \rightarrow 5 \rightarrow 7 \rightarrow 0 \rightarrow 6 \rightarrow 3 \rightarrow 6 \rightarrow 9$;
- Tino segue il percorso: $1 \rightarrow 2 \rightarrow 4 \rightarrow 0 \rightarrow 7 \rightarrow 5 \rightarrow 8$.

Si può dimostrare che non esiste alcuna pattuglia di solennità 3 o più.