

World directors (directors)

Souris Stofl et Souris Binna viennent d'être nommés comme les deux nouveaux *Directeurs du pays des souris*.

Le pays des souris est formé de N villes, numérotées de 0 à $N - 1$, et de $N - 1$ routes à double sens. Il est possible de se déplacer entre n'importe quelle paire de villes en utilisant une séquence de routes. La distance entre deux villes est le nombre minimum de routes qu'il faut traverser pour aller d'une ville à l'autre.

En tant que directeurs fraîchement nommés, Stofl et Binna doivent effectuer une *patrouille* traditionnelle de tout le pays des souris, qui se déroule comme suit :

- D'abord, Stofl et Binna se rendent dans des villes initiales X et Y .
- Ensuite, chaque jour, l'un des deux directeurs se déplace de sa ville actuelle vers une ville accessible par une seule route. Ça continue jusqu'à ce que chaque directeur soit passé par chaque ville au moins une fois et qu'ils soient retournés à leurs villes de départ. Note bien qu'un directeur peut se déplacer plusieurs jours de suite : les deux directeurs n'ont pas besoin de se déplacer à tour de rôle.

Stofl et Binna savent très bien que leur patrouille sera considérée comme plus solennelle s'ils sont éloignés l'un de l'autre : la *solennité* d'une patrouille est la distance entre les deux directeurs au moment où ils sont le plus proches.

Les deux directeurs t'ont donc engagé pour les aider à planifier leur patrouille et ton travail est de répondre à Q questions du type suivant :

- Si Stofl part de la ville X et Binna de la ville Y , quelle est la solennité maximale qu'ils peuvent atteindre avec leur patrouille ?

Implémentation

Tu dois soumettre un seul fichier avec l'extension `.cpp`.



Parmi les fichiers joints à ce problème, tu trouveras un modèle `directors.cpp` avec un exemple d'implémentation.

Tu dois implémenter les fonctions suivantes :

C++

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

- L'entier N est le nombre de villes.
- Les tableaux A et B , indexés de 0 à $N - 2$, contiennent les routes. En particulier, la i -ème route relie A_i et B_i .
- La fonction sera appelée une seule fois au début de l'exécution de ton programme.

C++

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

- Les entiers X et Y sont les villes de départ de Stofl et Binna.
- La fonction doit retourner la solennité maximale d'une patrouille où Stofl et Binna partent des villes X et Y .

- La fonction sera appelée Q fois pendant l'exécution de ton programme.

Évaluateur

Une version simplifiée du `< grader >` utilisé pendant la correction est disponible dans le dossier de ce problème. Tu peux l'utiliser pour tester tes solutions localement. Le `< grader >` d'exemple lit les données d'entrée depuis `stdin`, appelle la fonction que tu dois implémenter, et écrit le résultat dans `stdout` au format suivant.

Le fichier d'entrée est composé de $N + Q$ lignes, contenant :

- Ligne 1 : les entiers N et Q .
- Ligne $2 + i$ ($0 \leq i < N - 1$) : les entiers A_i et B_i .
- Ligne $N + 1 + j$ ($0 \leq j < Q$) : les entiers X_j et Y_j .

Le fichier de sortie est composé de Q lignes, contenant les valeurs retournées par la fonction `patrol`.

Contraintes

- $1 \leq N \leq 200\,000$.
- $1 \leq Q \leq 100\,000$.
- $0 \leq X, Y < N$ dans chaque requête.

Score

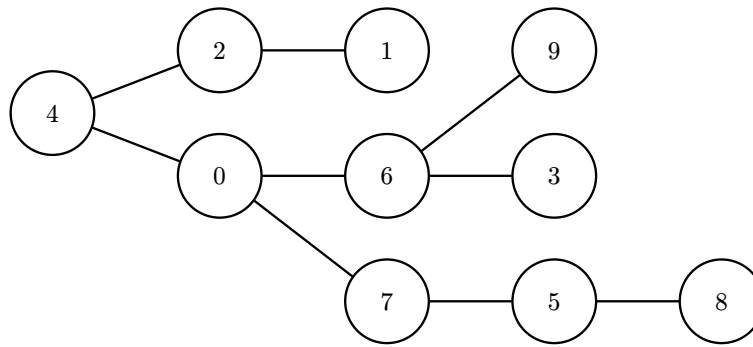
- **Sous-tâche 0 [0 points]**: Exemple.
- **Sous-tâche 1 [8 points]**: $A_i = 0$, $B_i = i + 1$ pour tout $0 \leq i < N - 1$.
- **Sous-tâche 2 [16 points]**: $A_i = 0$, $B_i = i + 1$ ou $A_i = i$, $B_i = i + 1$ pour tout $0 \leq i < N - 1$.
- **Sous-tâche 3 [13 points]**: $N, Q \leq 200$.
- **Sous-tâche 4 [14 points]**: $N \leq 1000$.
- **Sous-tâche 5 [18 points]**: Dans chaque requête, X et Y maximisent la réponse sur toutes les villes de départ possibles.
- **Sous-tâche 6 [17 points]**: $Q \leq 200$.
- **Sous-tâche 7 [14 points]**: Pas de contraintes supplémentaires.

Exemples

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	

Explication

Dans le premier cas d'exemple, le pays des souris a la structure suivante :



Pour la première requête, on peut atteindre une patrouille de solennité 2 avec les déplacements suivants :

- Stoffl prend le chemin : $9 \rightarrow 6 \rightarrow 3 \rightarrow 6 \rightarrow 0 \rightarrow 4 \rightarrow 2 \rightarrow 1$;
- Binna prend le chemin : $8 \rightarrow 5 \rightarrow 7 \rightarrow 0 \rightarrow 6 \rightarrow 3 \rightarrow 6 \rightarrow 9$;
- Stoffl prend le chemin : $1 \rightarrow 2 \rightarrow 4 \rightarrow 0 \rightarrow 7 \rightarrow 5 \rightarrow 8$;
- Binna prend le chemin : $9 \rightarrow 6 \rightarrow 3 \rightarrow 6 \rightarrow 0 \rightarrow 4 \rightarrow 2 \rightarrow 1$;
- Stoffl prend le chemin : $8 \rightarrow 5 \rightarrow 7 \rightarrow 0 \rightarrow 6 \rightarrow 3 \rightarrow 6 \rightarrow 9$;
- Binna prend le chemin : $1 \rightarrow 2 \rightarrow 4 \rightarrow 0 \rightarrow 7 \rightarrow 5 \rightarrow 8$.

On peut démontrer qu'aucune patrouille de solennité 3 ou plus n'existe.