# beCP

## 2019

# Tâche 2.2: Pourchassée par des monstres (monsters)

| Auteur: Jorik Jooken | Préparation: Jorik Jooken |
|---|---|
| Limite de temps: 2 s | Limite mémoire: 512 MB |

Vous recevez un graphe dirigé de $n$ nœuds et $m$ connexions. Ce graphe va changer après chaque étape. Ce que vous recevez c'est le graphe initial (à la première étape, l'étape 0). Après chaque étape, toutes les $m$ connexions sont remplacées : une connexion du nœud $x$ vers le nœud $y$ est remplacée par une connection du nœud $x$ vers le nœud $(y+s)\%n$. Le symbole % indique ici le reste après la division euclidienne. Par exemple, $10\%3 = 1$, $9\%7 = 2$ et $16\%7 = 2$.

Alice se trouve initialement au nœud $a$ et veut rejoindre Bob qui se trouve au nœud $b$. Il y a aussi $k$ monstres dans ce graphe. À chaque étape, Alice ainsi que chacun des monstres peut effectuer une action de type 1 ou 2. Type 1 : rester dans le nœud actuel. Type 2 : se déplacer sur un nœud adjacent (en suivant une connexion dans la bonne direction). Toutes les actions sont simultanées, et sont complétées avant que le graphe ne change. Une fois que le jeu démarre, Alice ne voit pas les monstres. Elle ne peut donc pas ajuster ses choix sur base de ce que les monstres ont fait aux pas de temps précédents.

Si Alice se trouve sur le même nœud qu'un monstre, le monstre mange Alice. Vous vous posez donc la question suivante : Alice peut-elle atteindre Bob strictement avant d'être mangée par un monstre ? Notez que si Alice arrive au nœud $b$ exactement au même moment qu'un monstre, ce dernier mange Alice (et dans ce cas, elle n'aura PAS atteint Bob strictement avant d'être mangée). Un monstre ne peut manger Alice que lorsqu'ils se trouvent tous les deux sur un nœud, et pas pendant qu'ils bougent d'un nœud à l'autre. De plus, un nœud peut contenir plusieurs entités (Alice, Bob ou des monstres) au même moment. Remarquez aussi que Bob NE peut PAS se déplacer.

Votre tâche est de déterminer la longueur de la plus petite suite (finie) d'actions qu'Alice doit exécuter afin de garantir qu'elle atteindra Bob strictement avant d'être mangée, ou bien de déterminer que trouver une telle suite d'actions est impossible. Une séquence d'actions $S_1$ garantit qu'Alice n'est pas mangée si et seulement si il n'existe pas de suite d'actions $S_2$ qu'un monstre peut effectuer afin de manger Alice avant que (ou exactement au moment où) elle arrive au nœud $b$.

## Input

La première ligne de l'entrée contient deux entiers $n$ et $m$ : le nombre de nœuds et le nombre de connexions. Les $m$ lignes suivantes contiennent chacune deux entiers $u_i$ $(0 \le u_i < n)$ et $v_i$ $(0 \le v_i < n)$. Ceci indique que dans le graphe tel qu'il est à l'étape 0, il y a une connexion allant du nœud $u_i$ vers le nœud $v_i$. La ligne suivante contient trois entiers $s$ ($s = 0$ ou $s = 1$), $a$ $(0 \le a < n)$ et $b$ $(0 \le b < n)$. Ici, $s$ indique comment le graphe change après chaque étape (voir l'énoncé du problème), $a$ est le nœud de départ d'Alice et $b$ est le nœud de Bob. La ligne suivante contient un entier $k$ : le nombre de monstres. Les $k$ lignes suivantes contiennent chacun un nombre $p_i$ $(0 \le p_i < n)$ : le nœud où le $i$-ème monstre se trouve initialement (au début de l'étape 0).

## Output

S'il n'y a pas de suite d'actions valide pour Alice qui garantit qu'elle peut atteindre Bob strictement avant d'être mangée par un monstre, imprimez $-1$ sur une seule ligne.

Sinon, imprimez la longueur de la plus courte suite d'actions qu'Alice doit effectuer afin de garantir qu'elle peut atteindre Bob strictement avant d'être mangée.

## Limites générales

- $1 \le n \le 2 \times 10^5$, le nombre de nœuds dans le graphe ;
- $0 \le m \le 2 \times 10^5$, le nombre de connexions dans le graphe ;
- $0 \le k \le 2 \times 10^5$, le nombre de monstres dans le graphe ;
- $s = 0$ ou $s = 1$

## Contraintes supplémentaires

| Sous-tâche | Points | Contraintes |
|---|---|---|
| A | 10 | $s = 0$ et $k = 0$ |
| B | 35 | $s = 0$ et $k = 1$ |
| C | 15 | $s = 0$ |
| D | 35 | $s = 1$, $k = 0$, $1 \le n \le 10^3$ et $0 \le m \le 10^3$ |
| E | 5 | $s = 1$, $k = 1$, $1 \le n \le 10^3$ et $0 \le m \le 10^3$ |

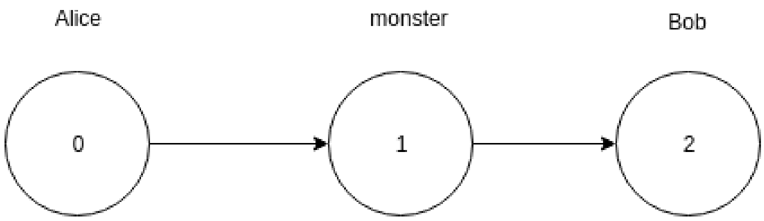
## Exemple 1

sample1.in
```
3 2
0 1
1 2
0 0 2
1
1
```

sample1.out
```
-1
```

La situation est illustrée dans la Figure 1. Alice se trouve initialement au nœud 0, le monstre est initialement au nœud 1 et Bob est au nœud 2. Dans cet exemple, $s = 0$, ce qui veut dire que le graphe reste identique après chaque étape. Il n'y a pas de suite d'actions qu'Alice peut effectuer afin d'atteindre Bob et éviter d'être mangée. Vous devez donc imprimer $-1$.

FIGURE 1 – Le graphe au début de l'étape 0, dans l'Exemple 1




## Exemple 2

sample2.in
```
3 2
0 2
2 0
1 0 1
0
```

sample2.out
```
2
```

Le graphe tel qu'il est au début de l'étape 0 est illustré dans la Figure 2. Alice se trouve initialement au nœud 0 et Bob au nœud 1. Il n'y a pas de monstres. Alice n'a qu'à suivre la connexion qui mène au nœud 2.

Comme $s = 1$, le graphe change à la fin de l'étape 0. Le nouveau graphe est illustré dans la Figure 3. Alice se trouve maintenant au nœud 2. Elle peut suivre la connexion qui mène au nœud 1 afin de rejoindre Bob. Au total, elle a effectué 2 actions (et il n'était pas possible d'en effectuer moins afin de rejoindre Bob). La réponse est donc 2.

FIGURE 2 – Le graphe au début de l'étape 0, dans l'Exemple 2

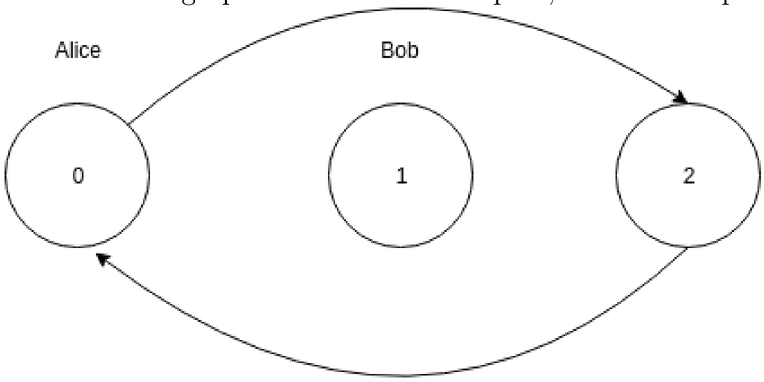
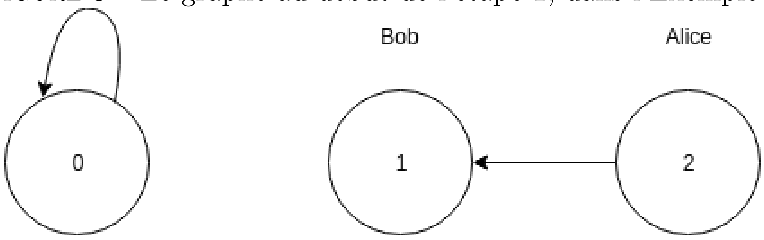



FIGURE 3 – Le graphe au début de l'étape 1, dans l'Exemple 2




### Exemple 3

sample3.in
```
4 4
0 1
1 2
3 0
3 1
0 0 2
1
3
```

sample3.out
```
-1
```

Le graphe tel qu'il est au début de l'étape 0 est illustré dans la Figure 4. Alice se trouve au nœud 0 et Bob au nœud 2. Il y a un monstre au nœud 3. Alice peut effectuer 2 actions (soit rester dans le nœud où elle se trouve ou bien suivre la connexion jusqu'au nœud 1). Si Alice reste où elle est, le monstre aurait pu choisir d'aller au nœud 0 et de la manger. Si Alice se déplace plutôt sur le nœud 1, le monstre aurait pu choisir d'aller au nœud 1 et de la manger quand même. Ainsi, il n'y a pas de suite d'actions pour Alice qui garantit qu'elle peut atteindre Bob strictement avant d'être mangée. La réponse est donc -1.

Figure 4 – Le graphe au début de l'étape 0, dans l'Exemple 3