

1 Présentation

Une devinette raconte l'histoire d'un berger qui possède un chou, une chèvre et un loup. En sa présence, la chèvre n'ose pas manger le chou, pas plus que le loup n'ose manger la chèvre, mais ils n'hésiteraient pas à satisfaire leurs appétits si l'homme tournait le dos. Ce berger doit traverser une rivière avec sa petite troupe et il ne dispose que d'une barque, dans laquelle il peut naviguer avec un seul de ses compagnons.

Peut-on résoudre cette devinette par un graphe ?

2 Résolution manuelle

2.1 États possibles

Activité 1 :

1. Énumérer les différentes configurations possibles.

rive de départ	rive d'arrivée
berger, chou, chèvre, loup	-
...	...

2. Éliminer les états qui ne répondent pas aux critères de la devinette.
3. Combien reste-t-il d'états ?

2.2 Graphe représentatif

La situation peut être représentée par un graphe qui recense par exemple les états possibles de la rive de départ. Une arête entre deux sommets exprime alors la transition de la rive de départ. Par exemple, la figure 1 montre que la rive de départ est passée de l'état *berger, chou, loup* à l'état *chou*. Dis autrement, le berger a emmené le loup sur l'autre rive.

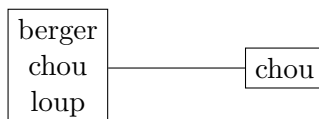


FIGURE 1 – Exemple de transition

Activité 2 : Construire le graphe des états de la rive de départ.

2.3 Solution

Activité 3 :

1. À partir du graphe, trouver une solution pour que le berger transporte tout le monde sur l'autre rive.
2. Existe-t-il un plus court chemin ?

3 Résolution informatique

3.1 Résoudre le problème

Activité 4 : En utilisant la classe *Graphe* construite en cours, retrouver et afficher le nombre de transition minimal que doit effectuer le berger.

3.2 Visualiser la solution

La bibliothèque *networkx* permet de construire mais également visualiser des graphes.

Activité 5 : En s'aidant de la documentation en ligne, utiliser la bibliothèque *networkx* pour réaliser le graphe résolvant la devinette.