

## Recherche dans des graphes d'états

### Exercice 1

Trois chèvres et trois loups se trouvent sur le même bord de la rivière. Il n'y a qu'un seul bateau qui peut contenir au plus deux animaux. Si les loups se retrouvent plus nombreux que les chèvres d'un côté ou de l'autre de la rivière, les chèvres se feront dévorées par les loups. Peut-on transporter sans risque de festin toutes les chèvres et tous les loups de l'autre côté de la rivière ? Attention, la barque ne peut faire la traversée à vide.

### Exercice 2

```
1 function Graph Search returns a solution or failure
2   initialise the frontier with initial state
3   initialise the explored set with {}
4   do
5     if the frontier is empty
6       then return failure
7     else
8       choose a node n and remove it from the frontier
9       if n is a goal state
10        then return n
11      else
12        add n to the explored set
13        for each successor s of node n
14          if s is not in the frontier or explored set
15            add s in the frontier
```

En vous inspirant du pseudo code de la fonction Graph Search ci-dessus, écrire le pseudo code de la fonction recherche

- en largeur d'abord
- en profondeur d'abord

### Exercice 3

On considère un arbre de recherche ayant les paramètres suivants :

- $b$  est le facteur de branchement de l'arbre
- $d$  est la profondeur de la solution la moins profonde
- $m$  est la profondeur maximale de l'arbre de recherche
- $l$  est la limite de profondeur de la recherche en profondeur limitée

Compléter le tableau ci-dessous en justifiant vos réponses.

	BFS	DFS	Depth-limited	Iterative-deepening	Bidirectional
complet ?	52 ? 56	52 ? 56	52 ? 56	52 ? 56	52 ? 56
complexité temps	$\mathcal{O}(?)$	$\mathcal{O}(?)$	$\mathcal{O}(?)$	$\mathcal{O}(?)$	$\mathcal{O}(?)$
complexité mémoire	$\mathcal{O}(?)$	$\mathcal{O}(?)$	$\mathcal{O}(?)$	$\mathcal{O}(?)$	$\mathcal{O}(?)$
optimal ?	52 ? 56	52 ? 56	52 ? 56	52 ? 56	52 ? 56