

## Série 2

## Exercice 1:

1. • DFS:

↳ ordre:  $S \xrightarrow{2} B \xrightarrow{3} D \xrightarrow{4} E \xrightarrow{5} G$

↳ chemin final:  $S \rightarrow B \rightarrow D \rightarrow E \rightarrow G$

↳ coût:  $1 + 5 + 1 + 4 = 11$

• BFS:

↳ ordre:  $S \xleftarrow{2} B \xleftarrow{3} C \xleftarrow{4} D \xleftarrow{5} E \xleftarrow{6} F \xleftarrow{7} G$

↳ chemin final:  $S \rightarrow B \rightarrow D \rightarrow G$

↳ coût:  $1 + 5 + 3 = 9$

• UCS:

↳ ordre:  $S \xleftarrow{2} B \xleftarrow{3} C \xleftarrow{4} D \xleftarrow{5} E$

↳ chemin final:  $S \rightarrow C \rightarrow D \rightarrow G$

↳ coût:  $2 + 2 + 3 = 7$

2. • DFS et BFS ne sont pas optimaux car ils ne prennent pas du tout en compte le poids de chaque chemin.
- UCS est selon moi optimal, car il choisit le chemin le moins coûteux.
  - Efficacité: les 3 algorithmes fonctionnent dans le cas donné. Mais avec des graphes plus longs et complexes, BFS et UCS peuvent prendre pas mal de temps (je ne suis pas sûre d'avoir bien compris la question)

## Exercice 3:

- States: les gares dans les différents cantons et les trajets possibles entre elles.
- Initial state: on est à la gare de Neuchâtel
- Goal state: être à la gare de Genève aéroport
- Action: prendre un train qui nous amène d'une gare à une autre
- Costs: le temps de trajet / prix du trajet

- J'ai choisi comme « coût » le temps de trajet (en minutes)

