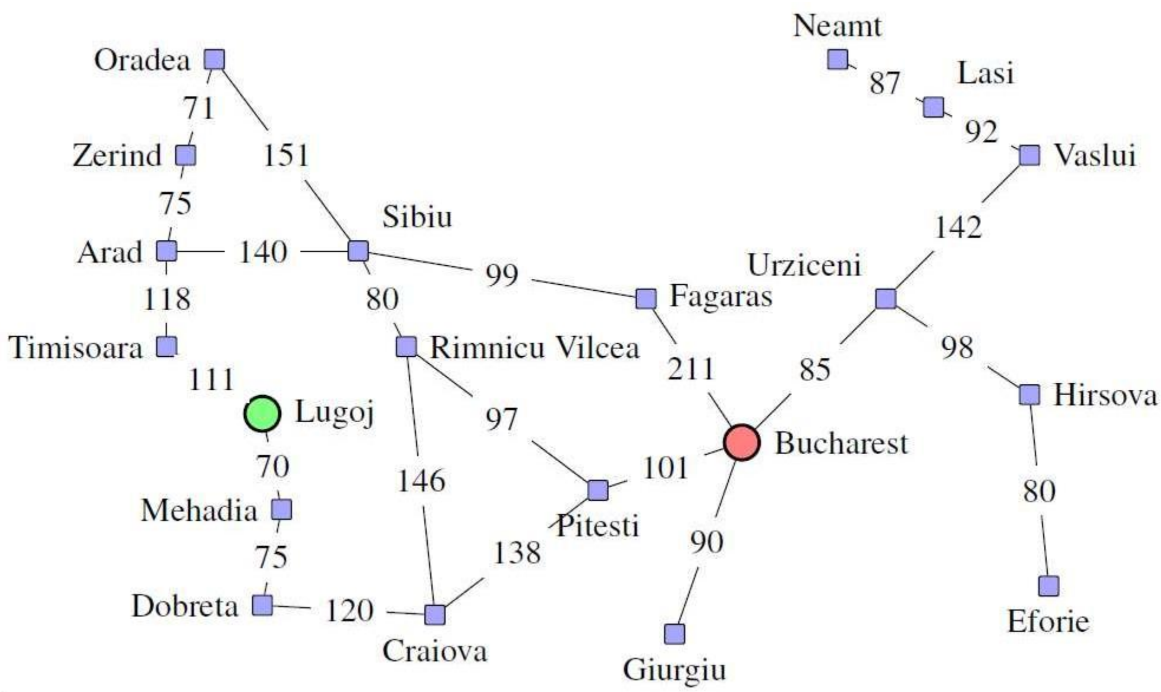
R4.04 : Méthodes d’optimisation
TD 2

Exercice 1 : L’algorithme A* :

Soit le graphe G et l’heuristique h(n) suivants :

On souhaite appliquez l’algorithme A * au problème du voyage en Roumanie en appliquant l’heuristique h(n) de la distance à vol d’oiseau dont les valeurs sont décrites dans le tableau suivant. La ville de départ est Lugoj et celle de destination est Bucharest.

Ligne droite jusqu’à Bucharest			
Arad	366	Mehadia	241
Bucharest	0	Neamt	234
Craiova	160	Oradea	380
Dobreta	242	Pitesti	100
Eforie	161	Rimnicu Vilcea	193
Fagaras	176	Sibiu	253
Giurgiu	77	Timisoara	329
Hirsova	151	Urziceni	80
Lasi	226	Vaslui	199
Lugoj	244	Zerind	374

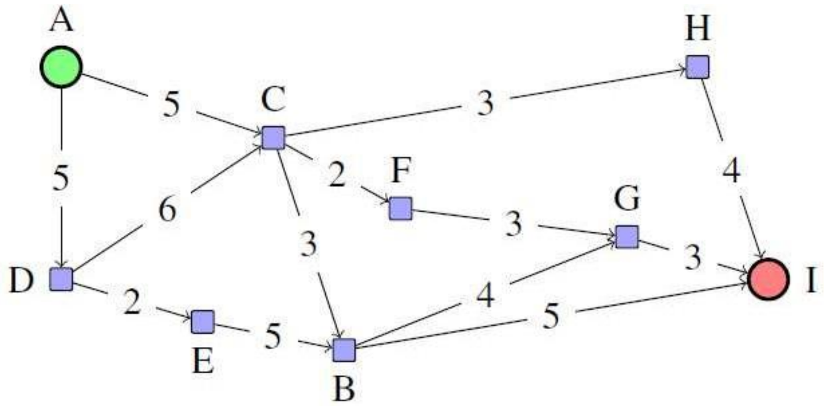


1. Simulez l'exécution de l'algorithme A* et donnez l'ensemble d'itérations avec le contenu des listes « Open » et « Closed » à chaque fois.
2. Donnez la solution finale de l'algorithme A*
3. L'heuristique h(n) est-elle admissible ? justifiez.

Exercice 2 : L'algorithme A* : Les heuristiques

Considérez la carte suivante. L'objectif est de trouver le chemin le plus court de A vers I. On donne également trois heuristiques, h1, h2 et h3.

Nœud	A	B	C	D	E	F	G	H	I
h_1	10	5	5	10	10	3	3	3	0
h_2	10	2	8	11	6	2	1	5	0
h_3	10	2	6	11	9	6	3	4	0



1. Est-ce que h1, h2 et h3 sont admissibles ? Justifier.
2. Est-ce que h4 = max(h1, h3) est admissible ? Justifier.
3. Appliquer la recherche A* en utilisant h1. Donner la solution finale.
4. Appliquer la recherche A* en utilisant h3. Donner la solution finale.
5. Appliquer la recherche A* en utilisant h4. Donner la solution finale.
6. Si vous avez le choix entre trois heuristiques admissibles h1, h2 et h3 = max(h1,h2) laquelle choisissez-vous ? Justifier.

Exercice 3 : Descente de gradient : le gâteau parfait

Imaginez que vous voulez optimiser la quantité de **sucre** (en grammes) dans une recette de gâteau pour maximiser le **niveau de satisfaction** des testeurs.

Problème :

La satisfaction des testeurs S(x) dépend de la quantité de sucre utilisée x, mais cette relation est plus complexe qu'une simple parabole. Elle est modélisée par une fonction polynomiale de degré 3 :

$$S(x) = -0.01x^3 + 0.6x^2 - 10x + 50$$

Consignes de résolution : écrire un programme Python dont :

Objectif : Trouver la quantité optimale de sucre x qui maximise la satisfaction S(x).

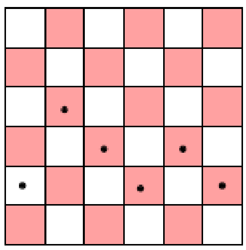
Données :

- x est la quantité de sucre (en grammes), et elle doit être comprise entre **0 g et 60 g**.
- S(x) est le niveau de satisfaction des testeurs (sur une échelle de 0 à 100).
- Utilisez la méthode de descente de gradient pour déterminer la valeur optimale de x.

Partie 3 : Problème des 6 reines, approche locale.

Résoudre le problème des 6 reines par l'algorithme de recherche locale (le problème consiste à placer 6 reines sur un échiquier 6*6 sans que deux d'entre elles ne se menacent mutuellement), en tenant compte du fait qu'il y a exactement une reine par colonne.

On commence avec les reines placées comme dans la figure ci-dessous. On utilise comme fonction d'utilité le nombre de paires de reines qui s'attaquent mutuellement. Dans la situation de départ il y a 9 paires de reines qui s'attaquent mutuellement. On choisit de déplacer une reine vers la case qui permet de réduire le plus possible ce nombre.



1. Codez la solution de recherche en python et donnez la solution finale