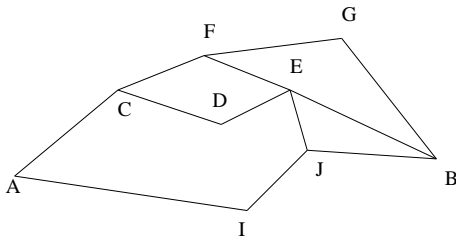


Algorithmes de recherche (A*, locale, génétique)

► **Exercice 1.** (Partiel 2005) Considérez la carte suivante avec des villes:



Les routes entre les villes peuvent contenir des parties montantes, descendantes et plates. Dans les parties montantes on peut avancer à 60 km/h, dans les parties descendantes à 120 km/h et dans les parties plates à 90 km/h. Dans le tableau suivant sont indiqués pour chaque route, le nombre de kilomètres du chemin entre deux villes par partie (montantes, descendantes, plates). Par exemple entre A et C, il y a 55 km (20 montants, 20 descendants et 15 plats).

Chemin entre	A et C	A,I	C,D	C,F	D,E	E,J	E,B	F,E	F,G	G,B	I,J	J,B
montante	20	50	10	20	0	0	20	0	50	0	0	10
descendante	20	10	20	0	10	0	30	20	0	60	10	10
plate	15	30	9	21	9	30	0	21	0	0	30	21

On veut trouver le chemin le plus court en **temps** (minutes) entre A et B. Pour cela on veut appliquer l'algorithme A*. On dispose en plus de l'information suivante: Pour chaque ville X on connaît la taille des parties montantes, descendantes et plates sur le chemin à vol d'oiseau entre la ville X et B.

Chemin directe entre	A et B	C,B	D,B	E,B	F,B	G,B	I,B	J,B
montante	40	30	30	40	40	20	20	10
descendante	40	30	20	0	10	30	0	10
plate	30	18	12	0	12	0	12	12

- L'heuristique associant à X le temps de parcours du chemin à vol d'oiseau de X vers B est-elle admissible ?
- Donnez une heuristique h **admissible**. Cette heuristique devrait être **en général** admissible pour des cartes similaires. Expliquez brièvement pourquoi elle est admissible. Donnez pour chaque ville la valeur h avec votre heuristique.
- Appliquez la recherche gloutonne (sans test d'états répétés) avec votre heuristique.
- Appliquez A* (sans test d'états répétés) avec votre heuristique.

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 ligne. Vous pouvez vous aider de l'échiquier fourni à la figure 1, en vous fabriquant des jetons. On commence avec les reines placées comme indiqué. 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.

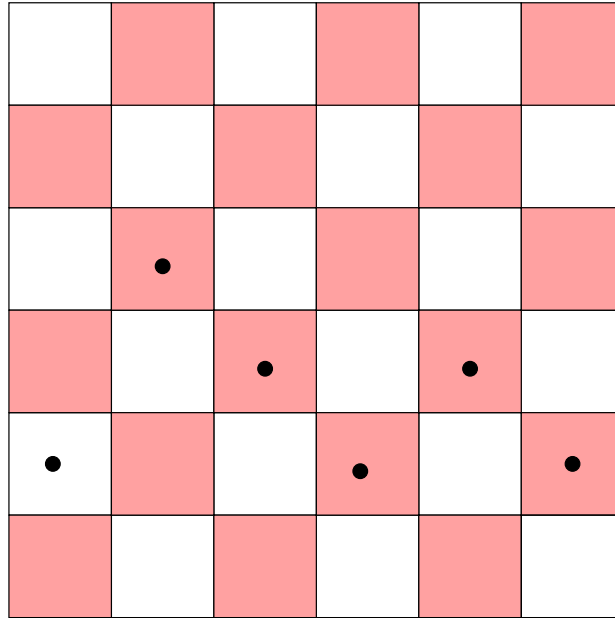


Figure 1: échiquier

► **Exercice 3.** On considère le problème du voyageur de commerce pour un graphe complet. On suppose que les villes sont numérotées de 1 à n . Le problème est de trouver le tour (circuit qui visite toutes les villes une fois) le plus court.

Pour utiliser l'algorithme génétique on doit coder le problème permettant de réaliser le croisement et la mutation.

Nous considérons le codage suivant : On écrit un "tour" comme une suite des n villes. On combine deux tours en les coupant de manière aléatoire, au même endroit, et en recollant les morceaux.

exemple avec $n = 5$:

parent 1	4	5	1	⋮	2	3	→	enfant 1	4	5	1	⋮	3	5
parent 2	4	2	1	⋮	3	5		enfant 2	4	2	1	⋮	2	3

Les mutations sont obtenues en inversant deux villes dans le tour.

- Est-ce que ce codage permet d'appliquer l'algorithme génétique ?
- Comment résoudre les problèmes qui se posent ?