# **TP2 – Heuristique pour le Voyageur de commerce**

## Heuristiques gloutonnes

* **Tournée croissante**

Question 1 : Donnez la tournée croissante pour les jeux de données

Jeu de donnée 1 :

* 3 ; 2 ; 6 ; 5 ; 4 ; 1

Jeu de donnée 2 :

* 5 ; 4 ; 2 ; 3 ; 6 ; 1

Question 2 : Déterminez la distance parcourue dans les deux cas.

Jeu de donnée 1 :

* Distance : 23

Jeu de donnée 2 :

* Distance : 14
* **Plus proche voisin**

Question 3 : Donnez la tournée obtenue pour les deux jeux de données.

Jeu de donnée 1 :

* 6 ; 2 ; 3 ; 6 ; 5 ; 4 ; 1

Jeu de donnée 2 :

* 2 ; 3 ; 6 ; 1 ; 5 ; 4 ; 1

Question 4 : Déterminez la distance parcourue dans les deux cas.

Jeu de donnée 1 :

* Distance : 22

Jeu de donnée 2 :

* Distance : 14

Question 5 : Quelles structures de données utiliseriez-vous pour coder cette heuristique

* **Insertion proche et loin**
  + **Distance d’un lieu L à un couple de lieu (A, B)**

Question 6 : Calculez la distance de l’usine aux deux premiers magasins dans le premier jeu d’essai.

LDistance(A,B) = A.distance(L) + L.distance(B) – A.distance(B)

Jeu de donnée 1 :

* 1 + 4 – 3 = 2

L = 1

A = 6

B = 5

Question 7 : Que représente cette distance

Cela correspond à la distance du lieu d’origine (L) par rapport aux deux distances (A,B).

* + **Distance d’un lieu L à une tournée T**

Question 8 : Calculez la distance de l’usine à la tournée composée des 4 premiers magasins du premier jeu d’essai

L.distance(T) = Min(L .distance(A,B),L.distance(B,C),L.distance(C,D), L.distance(D,A))

Jeu de donnée 1 :

* L.distance(A,B) = 6+3-2 = 7
* L.distance(B,C) = 3+1-2 = 2
* L.Distance(C,D) = 1+4-3 = 2
* L.Distance(D,A) = 6+8-9 = 5
* Min(7,2,2,5) = 2

L = 1

A = 3

B = 2

C = 6

D = 5

Question 9 : Que représente cette distance ? Maintenant que nous disposons de ces deux notions, nous pouvons nous en servir pour construire des tournées.

Cela représente la distance la plus proche dans la tournée que risque de faire le voyageur de commerce.

* **Insertion au plus proche**

Question 10 : Faites tourner cet algorithme sur un exemple avec votre enseignant.

Question 11 : Donnez le pseudo-code permettant de trouver les deux lieux les plus éloignés.

Question 12 : Donnez le pseudo-code permettant de trouver le lieu non visité le plus proche d’une tournée.

* + **Insertion au plus loin**

Question 13 : Faites tourner cet algorithme sur un exemple avec votre enseignant.

## Heuristique de recherche locale

* **Voisinage d’une tournée**

Question 14 : Listez tous les tournées voisines de la tournée (1,2,3,4,5,6).

Le principe de notre heuristique de recherche locale est le suivant :

1. On part d’une tournée T donnée par l’une des heuristiques précédentes.

2. On regarde si une des voisines de T est meilleurs que T.

3. Si on en trouve une, on remplace T par cette tournée et on recommence.

4. Si on n’en trouve pas, on s’arrête (et on renvoie T).

Question 15 : Faites tourner cette heuristique sur le premier jeu de données en partant de la tournée croissante.

Question 16 : Proposez des variantes de cette heuristique