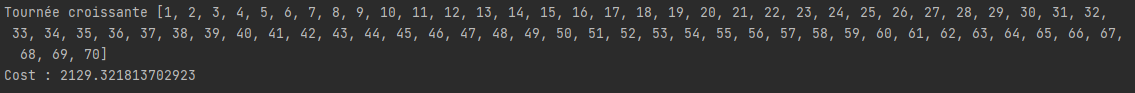
Séance TP3 Nom Prénom

Validation du TP1.

Question 1 : Coller en dessous un imprim écran de la tournée croissante et de son cout. Si le TP1 n’est toujours pas fini, expliquez-moi en détail où vous en êtes et ce qui vous bloque pour avancer.

Réponse :



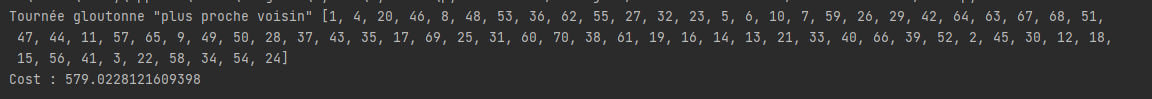
Validation du TP2

Question 2 : Avez-vous réussi à coder la méthode gloutonne plus proche voisin ? Si non pourquoi ?

Réponse :

Question 3 : Coller en dessous un imprim écran de votre résultat (la tournée plus\_proche\_voisin(1) + son cout)

Réponse :



Démarrage du TP3 : les recherches locales

Vous allez améliorer votre solution initiale de façon itérative, en cherchant dans le voisinage de la solution courante si une meilleure solution existe. Cette solution devient la nouvelle solution courante, dont on examine le voisinage. On répète le processus jusqu’à avoir une solution qui est meilleure que toutes ses voisines.

L’algorithme de base est le même pour chacune des recherches locales. La seule partie qui diffère est l’exploration du voisinage. Dans chaque fonction, on implémente un voisinage différent. Les principales difficultés portent sur la gestion du début et de la fin de la tournée, ainsi que sur la copie de la tournée voisine comme nouvelle tournée courante.

**Fonction recherche\_locale(tournée T\_entrée):tournée T\_sortie)**

Tcourante = T\_entrée

Fini=Faux

Tant que (Fini==Faux)

Fini=Vrai

Explorer le voisinage de Tcourante : //dépend du voisinage choisi

Si cout\_tournée(Tvoisin) < cout\_tournée(Tcourante)Alors

Tcourante=Tvoisin //nécessaire de faire une vraie copie

Fini=Faux

FinSi

Fin Tant Que

Retourner Tcourante

## Echange de successeurs

### Principe « premier d’abord »

Le voisinage est l’ensemble des tournées obtenues par inversion de deux villes consécutives dans la tournée. Dès qu’on trouve une amélioration, le voisin devient la tournée courante.

**Exploration\_successeurs\_premier\_d\_abord**

Pour toute position i dans Tcourante

Tvoisin = Echanger(Tcourante,i,i+1)

Autre façon de faire, en remplaçant la création de Tvoisin par le calcul direct du gain potentiel :

Pour toute position i dans Tcourante

Si d(i-1,i)+d(i+1,i+2) > d(i-1,i+1)+d(i,i+2) Alors

Tcourante = Echanger(Tcourante,i,i+1)

### Principe « meilleur d’abord »

Le voisinage est l’ensemble des tournées obtenues par inversion de deux villes consécutives dans la tournée. On attend d’avoir parcouru tout le voisinage pour choisir le meilleur voisin comme nouvelle tournée courante.

**Exploration\_successeurs\_meilleur\_d\_abord**

Pour toute position i dans Tcourante

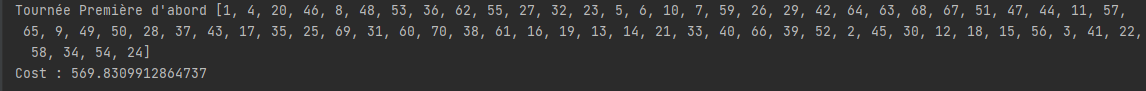
Tvoisin = Echanger(Tcourante,i,i+1)

Mémoriser le Meilleur\_voisin et le meilleur\_cout associé

Tcourante = Meilleur\_voisin

Question 4 : Choisissez un des deux principes (premier d’abord ou meilleur d’abord) et codez le. Testez ensuite en prenant comme solution de départ la tournée obtenue par plus\_proche\_voisin(1). Quel est votre résultat ? Tournée + cout de la tournée

Réponse :



## 

## Echange de sommets quelconques

### Principe « premier d’abord »

Le voisinage est l’ensemble des tournées obtenues par inversion de deux villes i et j quelconques dans la tournée. Dès qu’on trouve une amélioration, le voisin devient la tournée courante.

**Exploration\_ij\_premier\_d\_abord**

Pour toute position i dans Tcourante

Pour toute position j dans Tcourante

Tvoisin = Echanger(Tcourante,i,j)

### Principe « meilleur d’abord »

Le voisinage est l’ensemble des tournées obtenues par inversion de deux villes i et j quelconques dans la tournée. On attend d’avoir parcouru tout le voisinage pour choisir le meilleur voisin comme nouvelle tournée courante.

**Exploration\_ij\_meilleur\_d\_abord**

Pour toute position i dans Tcourante

Pour toute position j dans Tcourante

Tvoisin = Echanger(Tcourante,i,j)

Mémoriser le Meilleur\_voisin et le meilleur\_cout associé

Tcourante = Meilleur\_voisin

Question 5 : Choisissez un des deux principes (premier d’abord ou meilleur d’abord) et codez le. Testez ensuite en prenant comme solution de départ la tournée obtenue par plus\_proche\_voisin(1). Quel est votre résultat ? Tournée + cout de la tournée

Réponse :

## 

## Echange 2-opt

Une image contenant jeu

Description générée automatiquementLe voisinage est l’ensemble des tournées obtenues par inversion d’un tronçon de tournée, entre les villes i et j+1, comme indiqué sur le schéma ci-contre. On cherche ainsi à « démêler » les éventuels nœuds d’une tournée, en parcourant toute une partie de la tournée en sens inverse.

Là encore 2 possibilités:

* créer la tournée Tvoisine, calculer son cout, et remplacer Tcourante si le cout est meilleur
* calculer le gain potentiel du 2opt entre i et j et modifier Tcourante seulement si le gain est positif. Gain=d(i, i+1)+ d(j, j+1) -d(i,j)-d(i+1, j+1)puisque la partie de la tournée inversée garde la même longueur.

### Principe« premier d’abord »

**Exploration\_2opt\_premier\_d\_abord**

Pour toute position i dans Tcourante

Pour toute position j > i+1 dans Tcourante

Si d(i, i+1)+ d(j, j+1) > d(i,j)+d(i+1, j+1)

Tcourante = Retourner(Tcourante,i+1,j)

// on doit retourner toute la partie de tournée entre i+1 et j, pas seulement inverser les extrémités !

### Principe « meilleur d’abord »

**Exploration\_2opt\_meilleur\_d\_abord**

Pour toute position i dans Tcourante

Pour toute position j > i+1 dans Tcourante

Si d(i, i+1)+ d(j, j+1) > d(i,j)+d(i+1, j+1)

Mémoriser le Best\_cout, et les indices Best\_i Best\_j

Tcourante = Retourner(Tcourante,Best\_i+1,Best\_j)

Question 6 : Choisissez un des deux principes (premier d’abord ou meilleur d’abord) et codez le. Testez ensuite en prenant comme solution de départ la tournée obtenue par plus\_proche\_voisin(1). Quel est votre résultat ? Tournée + cout de la tournée

Réponse :

## 

## Variantes et Combinaison de Recherches locales

Les recherches locales implémentées peuvent être combinées tant que l’une d’elles apporte une amélioration à la solution courante.

Exemple d’algorithme :

**Ma\_super\_recherche\_locale(tournée T)**

Continer=True

Meilleur=T

Tant que Continuer

Continuer=False

T=recherche\_locale\_ij\_meilleur\_d\_abord(T)

T=recherche\_locale\_2opt\_premier\_d\_abord(T)

Si cout(Meilleur)<cout(T)

Meilleur=T

Continuer=True

FinSi

FinTant Que

En choisissant une autre tournée que glouton\_proche\_voisin(ville1), on peut obtenir de meilleurs résultats. Parfois c’est avec une moins bonne solution initiale que la tournée obtenue après recherche locale est meilleure. Les possibilités de combinaisons entre algorithmes gloutons et recherches\_locales sont multiples… Faites des essais !

Question 7 : Quel est votre meilleur résultat ? Tournée + cout de la tournée. Comment l’avez-vous obtenu ?

Réponse :