1. Qu'est-ce qu'une propriété NP ?

1. Les certificats sont des propositions de solution, justifiant la validité d'un algorithme et une réponse au problème.

   Pour notre problème Travelling Salesman Problem, notre certificat sera notre trajet possible pour répondre au problème … soit sur l'exemple le certificat renverra le tournée ACBDA.

   La taille d'un certificat corresponds aux nombres de villes à parcourir en plus du retour soit une ville de plus (n+1).

2.
   2.1. Voir l'implémentation de CertificatTSP. La fonction « suivant » va parcourir dans un mode exhaustive tous les certificats possible dans l'ordre qu'ils auront été générer.
   2.2. Il s'agirait d'un algorithme qui générerait un certificat aléatoirement et qui vérifiera sa validé (ou non).

3.
   3.1. On peut avoir au maximum n! certificats pour un problème ssi tous les parcours sont possibles.
   3.2. J'ai choisi de définir mes certificats par ordre de ville dans la matrice. (A puis B puis C …)
   3.3. Si l'algorithme arrive à générer un certificat, cela veut dire que le problème possède une solution.

<u>Implémentation</u>

Voir les classes TSP / CertificatTSP / NP / Certificat / testTSP – Terminé … OK

2. Réductions polynomiales

4.

4.1. Pour la réduction HamiltonCycle -> TSP. J'ai converti le tableau de booléen indiquant l'existence d'arête entre 2 villes par true par la distance minimal que l'on peut avoir soit 1 et dans le cas contraire pour false on aura une distance > 1.

4.2. Implémentation

Voir les classes : NPRed / HC / testHC – Terminé … OK

4.3. TSP est NP-dur, et donc NP-complet car on peut réduite HamiltonCycle qui est NP-complet en TSP.

4.4. TSP ne pourrait pas se réduire dans HamiltonCycle car TSP est connu NP-dur … cela viendrait a prouvé que HamiltonCycle est « au moins aussi dur » que TSP.

5.

5.1. Les données de HamiltonPath sont identique au donnée de HamiltonCycle … il y a unique que la sortie qui es différente avec un retour à la source pour HamiltonCycle.

5.2. Implémentation

Voir les classes : NPRed / HP / testHP – Terminé … OK

5.3. HamiltonPath peut être réduit en HamiltonCycle et HamiltonCycle en TSP donc par réduction en chaine on peut réduire HamiltonPath en TSP.

6. Implémentation

Voir la classe : testHP – Terminé … OK

3. Optimisation versus Décision

7. On peut dire que TSP peut être réduit facilement pour tous « l » obtenue inférieur ou égal à celui obtenue par les algorithmes TSPOpt1 et TSPOpt2. Et si TSPOpt1 et TSPOpt2 était P alors TSP le serait aussi.

8. Si TSP est P et qu'on arrive à réduire TSPOpt1 en TSP alors TSPOpt1 sera P également ssi une instance de TSPOpt1 peut être convertie en TSP avec l >= TSPOpt1 et la matrice de distance et le nombre de ville.

9. Si TSP est P et qu'on arrive à réduire TSPOpt2 en TSP alors TSPOpt2 sera P également ssi une instance de TSPOpt2 peut être convertie en TSP avec l >= TSPOpt2 + D[tour(n-1),tour(0)] et la matrice de distance et le nombre de ville.