TP –La classe NP

**PARTIE 1**

Q1)  
  
Définition de certificat :   
Le certificat est une donnée ajoutée au problème qui permet de spécifier si la réponse au problème est oui ou non.

C'est-à-dire que le certificat devra, dans cet exemple, dire quel Touré est le plus courte de villes.

Implémentation :   
Pour ce problème, le certificat sera donc représenté en un tableau d'entiers de taille n. Les valeurs du tableau seront les numéros des villes.

Taille d'un certificat :

n

Certificat est-il borné ? :

La taille de la donnée est :

-booléen vérifier doublons. Temps de calcule : O(n)

-Dist = (Som.D (i, i+1) + D (n-1, 0)). Temps de calcule : O(n)

-Vrai si Dist <= Valeur. Temps de calcule : O(1)

Complexité totale : O(n)+ O(n)+ O(1) = O(n)

Taille de certificat est : n

Est-ce que n est polynomial par rapport à O(n) ?

Toujours vrai. n est polynomial (polynôme de degré 1).

Le certificat est de taille polynomiale, l’algo de vérification est de complexité polynomiale le problème est donc NP.

Q2)

Q 2.1) Génération aléatoire d’un certificat :

Aucune idée

Q 2.2) Quel serait le schéma d’un algorithme non-déterministe polynomial pour le problème :

Le schéma d'un algorithme non déterministe est un algorithme qui donne un certificat au hasard, et il le vérifie, on joue donc en quelque sorte sur la chance.

Q3)

3.1 Pour n fixer les valeurs qui peuvent-prendre un certificat:

n!

3.2 Enumération de tous les certificats :

**Donné : 0--------------------------------n-1**

* Comparaison x(i) avec x (i-1) :

Si x(i) plus grand que x (i-1) on change la position

Règle pour Changer position :

Parcourir tous la donnée qui sont à droite de x (i-1) pour

Trouver les données qui sont supérieur a x (i-1) et

Prend la plus petite entre eux on le met à la place de x (i-1) et

Le reste en va le trier à l’ordre croissant

**Exemple :**

|  |
| --- |
| 1 2 **3** **4** |
| 1 2 4 3 |
| 1 3 2 4 |
| 1 3 4 2 |
| 1 4 2 3 |
| 1 4 3 2 |

**PARTIE 2**

Q4)

Q4.1)Hamilton Cycle se réduit polynomialement dans TSP :

Les sommets des graphes sont les villes

Il y a un arc entre le deux sommet s’il y a une une distance

Q4.2)

Q 4.3) Qu’en déduire pour TSP :

\_En supposent que NP\_complet et NP\_dure

HamiltonCycle est NP\_complet donc c’est un NP\_dure et il veut se réduit polynomial dans TSP

Ce a dire en peut dire que ->> TSP et NP\_dure

Q4.4 Pensez-vous que TSP se réduise polynomialement dans HamiltonCycle ?

Oui on peut faire une réduction dans l’autre sens :

TSP se réduit polynomialement en HamiltonCycle puisque que TSP est NP et HamiltonCycle NP-dur

5.1) Hamilton path (HP) ≤p Hamilton cycle (HC)

Le problème Hamilton possède N+1 nombre de sommets par rapport à un problème Hamilton path et Possède deux arrête de plus par rapport à un problème Hamilton path.

Hamilton path (HP) ≤p Hamilton cycle (HC)

|  |  |  |
| --- | --- | --- |
| Probleme Hamilton path |  | Probleme Hamilton Cycle |
| G(n, E)  depart  arrive | n’=n+1 | G’(n’,E’) |

Transformation des graphes :

Problème Hamilton Path Problème Hamilton Cycle

V’ = V + 1

Preuve :

Supposant en commence par la premier graphe en peut dire si en relis le des sommet par une autre sommet (f) en obtient la deuxième graphe et dans le sens inverse si en commence par la deuxième graphe et en supprime une sommet en obtient la premier graphe notion bien une sommet est relier par deux arc donc n-1 .

Dans ce cas en peut distinguer que l’algorithme de transformation est borné par la taille de l’entrée du problème Hamilton Path, donc l’algorithme est polynomial en O(n²).

5.3) Montrer que Hamilton Path se réduit dans TSP :

On sait que (HC ≤p TSP) et (HP ≤p HC)

En conclure que HP ≤p TSP car la réduction est transitive.

Donc HP ≤p HC ≤p TSP.

**PARTIE 3**

Q 7. Montrer que si TSP Opt1 (resp. TSPOpt1 Opt2) était P :

De ce fait TSP serait P aussi.

Si le problème TSPOpt1 était P, TSPOpt1 pourrait l'utiliser en mettant toutes les taille (length) à la même valeur donc, celui-ci serait P puisqu'il ne s'occuperait que de faire un appel à un problème P. Pour ce qui est de l'affichage de la solution, on pourrait modifier BinOpt1 pour renvoyer l'affichage ce qui ne changerait pas la complexité de l'algorithme.

De ce fait, TSP serait P.

8)

Aucune idée

9)

Si TSP est P, comme vu a la question précédent, TSPOpt1 aurait été P aussi.d'où, TSPOpt2 consisterait donc juste à appeler TSPOpt1 puis TSP qui seraient tous les deux P.Par conséquent, TSPOpt2 serait aussi P.