**TP1**

**Complexité et Heuristique**

1. **Complexité algorithmique**

Question 1 : Redonnez la définition de complexité d’un algorithme (dans le pire des cas).

Complexité = Estimation du nombre d’opération élémentaire effectués dans le pire des cas en fonction de la taille de l’entrée (n) pour résoudre un problème

Question 2 : En vous aidant d’Internet, donnez une définition simple de la notion de complexité d’un problème.

Complexité en temps minimale d’un algo qui résout le problème.

Question 3 : En vous aidant d’Internet, donnez une définition simple de la notion de classe de complexité et en particulier des classes T, P et NPc.

Classifier problème en fonction de leurs difficultés de résolutions.

**T :** Temps Constant quelle que soit la taille de l’entrée

**P :** Polynomiale (il existe un algo en temps polynomiale)

**NP complet :** Non déterministe polynomial (il n’existe pas d’algo capable de résoudre rapidement n’importe quelle instance du problème).

Question 4 : Cherchez des exemples de problèmes pour chacune de ces trois classes de complexités.

**T : Temps Constant quelle que soit la taille de l’entrée**

* Test de parité
* Renvoyer première valeur d’un tableau

**P : Polynomiale**

* Algo de tri (tri en fusion, tri par insertion, etc…)
* Le plus court chemin

**Npc : Non déterministe polynomial**

* Problème de coloration de graphe
* Problème ensemble dominant
* Problème voyageur de commerce

Question 5 : Quel est l’intérêt de connaître la complexité d’un problème avant de coder un programme pour le résoudre ? Expliquez le comportement à adopter en fonction de la classe de complexité du problème.

Cela nous permet de ne pas travailler pour rien, cela va changer notre approche du problème selon sa complexité.

L’on va donc s’adapter à la classe du problème.

1. **Heuristique et problème de coloration**

Question 6 : Donnez une définition simple d’une heuristique.

Algo produit rapidement une solution réalisable mais pas forcément la plus optimale.

Question 7 : Donnez le nom et le principe de l’heuristique de coloration de graphe que vous connaissez.

**Coloration :** DSATUR (degrés de saturation) en **n2** ou RLF **n3**

**Question 8 : Selon vous, qu’est-ce qu’une bonne heuristique et qu’est-ce qu’une mauvaise heuristique ?**

Bonne heuristique :

* Elle donne une réponse souvent de bonne qualité
* Elle est rapide / peu gourmande en mémoire

**Durant votre formation, vous allez être confronté principalement à trois types d’heuristiques :**

**• Des heuristiques gloutonnes**

**• Des heuristiques de recherche locale**

**• Des heuristiques génétiques**

**Question 9 : Expliquez brièvement le principe de chacune de ces trois types d’heuristiques.**

**Question 10 : Pour chacun de ces types d’heuristique, imaginez plusieurs exemples pour résoudre le problème de coloration d’un graphe.**

**Question 11 : Choisissez avec votre enseignant un problème de la classe NPc (autre que le voyageur de commerce et la coloration) et adaptez vos différentes idées d’heuristiques à ce nouveau problème.**

**Question 12 : Proposez des idées d’heuristiques pour le problème du voyageur de commerce**

d