

## UE: ALGORITHMIQUE APPLIQUÉE - 4TIN903U

(test – durée 1h)

– *Documents (cours et notes personnelles) autorisés* –

---

### Des reines sur un échiquier

Un problème classique consiste à placer huit reines d'un jeu d'échecs sur un échiquier de  $8 \times 8$  cases sans que les reines ne puissent se menacer mutuellement, conformément aux règles du jeu d'échecs (la couleur des pièces étant ignorée). Ainsi, deux reines ne devraient jamais partager la même rangée, colonne, ou diagonale de l'échiquier.

**Question 1.** *Un algorithme exhaustif naïf qui consisterait à explorer tous les positionnements des 8 reines sur l'échiquier devrait explorer combien de possibilités ?*

**Question 2.** *Expliquer comment modéliser ce problème par un problème de graphe à l'aide d'un graphe  $G$  possédant comme sommets les 64 cases de l'échiquier. Décrire les arêtes de  $G$  et spécifier le problème à résoudre.*

On propose la variante consistant à déterminer le nombre minimum de reines nécessaires pour contrôler toutes les cases d'un échiquier de taille  $n \times n$ .

**Question 3.** *Expliquer comment modéliser ce problème par un problème de graphe.*

**Question 4.** *Trouver le nombre minimal de reines à positionner pour contrôler un échiquier  $4 \times 4$ . Dessiner une solution et argumenter pour démontrer l'optimalité de votre solution.*

### Couvrir des points dans l'espace

On considère le problème PLANCOVER (couverture de points par des plans) défini comme suit :

**Entrée :** Un ensemble  $P \subset \mathbb{R}^3$  de  $n$  points de l'espace, et un entier  $k$ .

**Question :** Existe-t'il un ensemble de  $k$  plans passant par tous les points de  $P$  ?

Si nécessaire, vous supposerez connue la routine qui, étant donné un ensemble  $S$  d'au plus trois points, renvoie un plan passant par tous les points de  $S$ , de même que la routine permettant de tester si un point appartient à un plan ou une droite donné. On ne demande donc pas d'écrire ces routines.

On rappelle qu'il n'y a qu'un seul plan passant par trois points non-alignés, et que deux plans non-parallèles se coupent selon une droite.

**Question 5.** *Décrire un algorithme (et sa complexité) permettant de résoudre PLANCOVER lorsque  $k = 1$ . (Attention !  $P$  pourrait contenir plus de trois points alignés.)*

On suppose maintenant que  $P$  ne contient pas plus de  $t$  points alignés. Dit autrement, toute droite de l'espace passe par au plus  $t$  points de  $P$ .

**Question 6.** *Donner une règle de réduction montrant que PLANCOVER, lorsque  $k$  et  $t$  sont des paramètres, admet un noyau. Vous devez justifier, c'est-à-dire montrer que votre réduction est correcte et polynomiale tout en donnant la taille du noyau.*

**Question 7.** *En combinant la question précédente à un algorithme d'approximation, proposer un algorithme polynomial qui : soit répond "non" et alors  $P$  ne peut pas être couvert par  $k$  plans ; soit renvoie un ensemble de  $O(k \log(tk))$  plans couvrant tous les points de  $P$ . (Faites attention que même après le calcul du noyau, la couverture de  $P$  par  $O(k \log(tk))$  plans peut ne pas exister et dans un tel cas l'algorithme devra répondre "non".)*

## Décomposition arborescente des graphes planaires

On propose d'étudier des graphes planaires construits par la procédure suivante :

1. Construire une "étoile", un graphe composé d'un sommet central connecté à tous les autres (qu'on appellera feuilles) ;
2. Tant que le graphe reste planaire, ajouter des arêtes arbitrairement entre les feuilles.

**Question 8.** *Dessiner deux graphes planaires différents ayant chacun 7 sommets construits selon cette procédure. (Vous devez argumenter pour affirmer que ce sont bien des graphes différents même à une renumérotation près des sommets.)*

**Question 9.** *Démontrer que tous les graphes construits de cette manière sont de largeur arborescente au plus 3.*

**Question 10.** *Donner un exemple de tels graphes de largeur arborescente exactement 3.*

**FIN.**