**Epreuve finale : Algorithmique avancée et Complexité**

**Exercice 1 : (10 pts)**

Considérons le problème de compression des données. Soit en entrée une matrice binaire A de dimension (NxM). Le processus de compression construit un vecteur d’entiers T qui vérifie les propriétés suivantes :

* T[1] = N ; nombre de lignes de la matrice
* T[2] = M ; nombre de colonnes de la matrice
* T[3] = A[1, 1]. Le premier élément de la matrice
* Pour tout i > 3 et <=k, T[i] représente la longueur de la séquence de nombre identique
* K représente la dimension du tableau à construire
* Une séquence de 0 s’arrête lorsqu’un 1 est rencontré et vice versa.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 1 | 1 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 1 |

Cette matrice sera représenté par :

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 5 | 3 | 1 | 2 | 4 | 3 | 1 | 2 | 7 | 1 |

1. A quoi correspondent le meilleur cas et le pire cas pour ce problème de compression ? Justifier votre réponse. **(2 pts)**
2. Ecrire un algorithme permettant la compression d’une matrice binaire quelconque (N, M) Calculer la complexité temporelle et spatiale **(4 pts)**
3. Ecrire un Algorithme permettant de reconstituer la matrice M. Calculer la complexité temporelle et spatiale **(4 pts)**

**Exercice 2 : (10 pts)**

Considérer le problème de construction des sous ensemble de même somme à partir d’un ensemble d’entiers en entrée. Le problème consiste à construire K sous ensemble disjoints (aucune intersection entre les sous ensemble) telle que les sommes des éléments de chaque sous ensemble soient égaux. La définition formelle du problème est comme suit :

***Instance :*** Étant donné un ensemble *E* de *n* entiers positifs, et une somme cible S

***Question :*** Existe-t-il K sous-ensemble de *E* dont la somme des éléments de chaque sous ensemble esst égale à S ?

**Exemple :** Soit E = {80, 73, 17, 100, 90, 15, 21, 19, 45, 70, 120, 60, 35, 110, 85}, et S = 200, k = 3

1. Illustrer sur l’exemple les étapes de construction d’une solution en spécifiant la modélisation la plus adéquate **(3 pts)**
2. Estimer approximativement la taille de l’arbre de résolution et en déduire l’ordre de complexité de l’algorithme de résolution **(2 pts)**
3. Proposer un algorithme de validation d’une solution donnée S’et calculer sa complexité. **(4 pts)**
4. En déduire la classification associée au problème étudié. Justifier votre réponse **(1 pt)**