Algorithmique
TD n° 11 : Algorithmes gloutons et autres

Exercice 1 :

Soit `T[1..n]` un tableau d'entiers (positifs ou négatifs mais pas 0) contenant au moins un entier positif. Un sous-tableau contigu `T[i..j]` est appelé *intervalle positif*, si la somme des entrées du sous-tableau est strictement plus grand que 0. On s'intéresse au nombre minimal d'intervalles positifs qui couvrent chaque entrée positive du tableau. Par exemple pour le tableau `T = [-3,+3,-5,+7,-4,+1,-8,+3,-7,+5,-9,+5,-2,+4,-42]` ce nombre est 3 car on peut couvrir les entrées positives avec `T[2..6]`, `T[8..10]` et `T[12..14]`.

1. Proposez un algorithme glouton.

2. Montrez que l'algorithme ne trouve pas forcément le bon nombre minimal.

3. Donnez un algorithme qui trouve la solution optimale. Appliquez l'algorithme sur le contre-exemple trouvé précédemment. Quel est sa complexité ?

Exercice 2 :

On considère deux tableaux d'entier positifs ou nuls `L[1..n]` et `C[1..m]`. Une matrice M d'entiers positifs ou nuls de dimension $n \times m$ est appelée *compatible* avec L et C, si la somme des éléments de chaque ligne i est égale à $L[i]$ et que la somme des éléments de chaque colonne j est égale à $C[j]$. Un exemple d'une matrice compatible avec L et C est donné dans la figure 1.

L : \ C :	27	94	42	50
57	12	42	3	0
47	3	10	31	3
71	2	31	4	34
38	10	11	4	13

L : \ C :	20	15	30	5
22	?	?	?	?
13	?	?	?	?
4	?	?	?	?
31	?	?	?	?

FIGURE 1 – Une matrice compatible et un exemple à résoudre

1. Montrez que si M est compatible avec L et C, alors L et C doivent vérifier la propriété :

$$\sum_{i=1}^{n} L[i] = \sum_{j=1}^{m} C[j] \tag{1}$$

2. Montrez que si L et C satisfont la propriété (1), alors il existe une matrice M compatible avec L et C. Pour cela donnez un algorithme glouton qui construit une matrice M compatible avec L et C. Appliquez l'algorithme sur l'exemple de la figure 1.

Exercice 3 :

On considère deux tableaux d'entier positifs ou nuls `L[1..n]` et `C[1..m]` comme dans l'exercice précédent. Ici, on cherche une matrice *binaire* (c'est-à-dire composée de 0 et de 1) compatible avec L et C.

1. La propriété (1) de l'exercice précédent est toujours vraie pour chaque matrice binaire compatible. Par contre, même si on exige que $L[i]$ (resp. $C[j]$) soit toujours inférieur ou égal à n (resp. m), il y a des L et C vérifiant la propriété pour lesquels il n'y a pas de matrice binaire compatible. Donnez un exemple.

2. Proposez une propriété sur L et C plus forte de sorte qu'on soit sûr de pouvoir trouver une matrice binaire compatible. Pour cela donnez un algorithme glouton.

Exercice 4 :

Vous devez ranger tous les n livres d'une bibliothèque dans des étagères. Les étagères sont composées de tablettes qui peuvent contenir des livres jusqu'à une certaine hauteur. L'ordre des livres est fixé et ne peut pas être changé. Chaque tablette doit contenir un intervalle contigu de la séquence des livres. Il y a deux tableau `H[1..n]` et `E[1..n]` qui contiennent les hauteurs et les épaisseurs des livres dans l'ordre. Toutes les tablettes ont la même longueur L. L'épaisseur totale de tous les livres sur une tablette ne peut pas excédée L.

1. Supposons que tous les livres ont la même hauteur h et que toutes les tablettes peuvent stocker des livres de cette hauteur. Donnez un algorithme glouton pour stocker les livres en utilisant le moins de tablettes possibles. Montrez qu'il est correcte.

2. Supposons maintenant que les livres peuvent avoir des hauteurs différentes et qu'on peut ajuster chaque tablette pour accueillir le livre le plus haut sur cette tablette. En particulier, la hauteur de chaque tablette vide peut être 0. Le problème est de stocker les livres de sorte que la somme totale des hauteurs des tablettes soit le plus petit possible. Montrez que l'algorithme glouton précédent ne donne pas toujours la meilleur solution.

3. Donnez un algorithme pour trouver la meilleur solution.