

Introduction à la compression de données

Introduction

La compression de données (sans pertes) est une méthode qui consiste à diminuer la taille des données tout en nous permettant de les récupérer au complet. Par exemple, si vous avez plusieurs boîtes en carton de taille variable, une façon de les "compresser" pourrait être de les mettre les unes dans les autres. Un autre moyen pourrait être de les plier. Ces deux méthodes de rangement de boîtes sont différentes, mais permettent toutes les deux de gagner de la place tout en nous permettant de récupérer les boîtes initiales quand on le souhaite.

Pour ce travail, nous n'allons pas compresser des boîtes, mais des **messages**. À partir de maintenant, nous considérons qu'un message est un **tableau d'entiers**. La méthode de compression utilisée ici est le codage par répétitions.

Le codage par répétitions consiste à remplacer toute séquence de nombres identiques par le **nombre d'occurrences** du nombre suivi du nombre lui-même. Ainsi, par exemple, le message {1,1,1,0,0,3,3,3,3,3,3} sera compressé en {3,1,2,0,6,3}. En effet, dans le message original, on remarque qu'il y a d'abord 3 fois le nombre 1, puis 2 fois le nombre 0, puis 6 fois le nombre trois.

Vous remarquez que sur cet exemple, la version compressée prend moins de place que la version non-compressée. À votre avis, est-ce toujours le cas ?

1 Travail

On vous demande d'écrire un programme C permettant de compresser n'importe quel tableau de nombres.

Le programme contiendra une méthode main, où l'utilisateur est invité à entrer un tableau d'entiers. Pour ce faire, il entre d'abord la taille du tableau, puis chacun des entiers un par un. Le programme affichera par exemple :

```
Veillez indiquer la taille du tableau à compresser: 10
Nombre 1: 2
Nombre 2: 2
Nombre 3: 2
Nombre 4: 1
Nombre 5: 10
Nombre 6: 10
Nombre 7: 10
Nombre 8: 10
Nombre 9: 0
Nombre 10: 0
Tableau compressé: [3,2,1,1,4,10,2,0]
```

La fonction de compression à implémenter correspond à la signature et aux spécifications suivantes :

```
1 /*
2  * PRE: message est initialisé.  tailleMessage contient la taille du tableau message
```

```
3  * POST: affiche a l'ecran le message compresse et renvoie sa taille.  
4  *      message n'a pas ete modifie  
5  * */  
6  
7  int compresser(int message[],int tailleMessage);
```

2 Variante avec une file

Réalisez le même projet, mais en utilisant une **file** (Queue) au lieu d'un tableau.