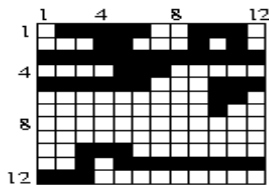


## Projet d'intégration PIM

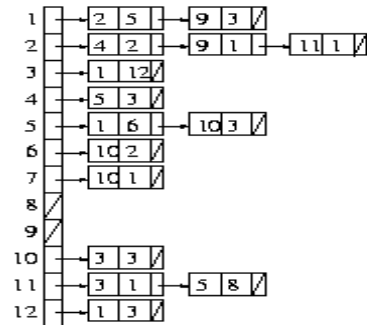
### Le problème

La transmission d'un document par FAX se fait en transformant chaque page du document en image noir et blanc, envoyée ensuite vers le FAX récepteur. Pour simplifier, l'image d'une page est une matrice de points. Chaque point peut être noir ou blanc.

On compresse l'image pour réduire son volume et accélérer sa transmission (voir figure ci-dessous comment se fait la compression de l'image):



a. Image en format simple



b. Image en format compressé

- 1- Ecrire un programme **Simuler\_Fax** qui fait la saisie d'une image en format simple de **dimensions quelconques choisies par l'utilisateur**, la convertit en format compressé, et affiche l'image à partir des formats simple et compressé (voir le déroulement du programme à la fin de ce document).

On utilisera le code suivant : **Point Noir = 1 ; point Blanc = 0.**

Pour cela :

- 2- Définir les bons types et les structures de données pour représenter une image sous les deux différents formats, simple et compressé.
- 3- Ecrire le programme qui demande à l'utilisateur de lire une image à partir de la ligne de commande ligne par ligne ou à partir d'un fichier.

### Cas 1 :

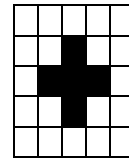
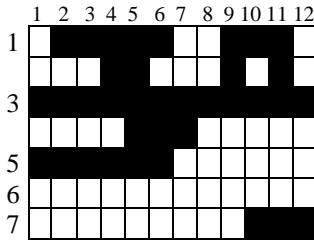
- 4- Ecrire une procédure qui demande à l'utilisateur le nombre **L** de lignes et le nombre **C** de colonnes (ou le nombre **C** de caractères par ligne) d'une image en format simple.
- 5- Ecrire une procédure qui saisit une image en format simple, (ligne par ligne, les "1" et les "0" sont espacés voir exemple 1).
- 6- Ecrire une procédure qui affiche une image à partir de son format simple.
- 7- Ecrire une procédure qui transforme l'image du format simple en format compressé.
- 8- Ecrire une procédure qui affiche une image à partir de son format compressé.
- 9- Adapter le programme pour que l'utilisateur puisse ressaisir une autre image sans quitter l'application.
- 10- Adapter la question 3 pour que le système redemande la saisie d'une ligne s'il y a une erreur de saisie dans cette ligne (voir l'exemple 2)

## Cas 2

- 11 Ecrire une procédure qui va lire une image à partir d'un fichier en format .png
- 12 Ecrire une procédure qui va sauvegarder l'image compressée en format .comp
- 13 Utiliser un outil pour visualiser votre image en format .png

**Conseil :** Tester au fur et à mesure chaque fonctionnalité demandée.

Deux exemples de déroulement du programme :



**Exemple 1 :** déroulement normal

**Exemple 2 :** Début de déroulement, erreur de saisie à la ligne 3

```

*****
* Bienvenu sur le simulateur de Fax *
*****
Entrer le nombre de lignes L de votre image : 7 ↵
Entrer le nombre de colonnes C de votre image : 12 ↵

Saisie de l'image normale
-----
Entrer la ligne 1 : 0 1 1 1 1 1 0 0 1 1 1 0 ↵
Entrer la ligne 2 : 0 0 0 1 1 0 0 0 1 0 1 0 ↵
Entrer la ligne 3 : 1 1 1 1 1 1 1 1 1 1 1 1 ↵
Entrer la ligne 4 : 0 0 0 0 1 1 1 0 0 0 0 0 ↵
Entrer la ligne 5 : 1 1 1 1 1 1 0 0 0 0 0 0 ↵
Entrer la ligne 6 : 0 0 0 0 0 0 0 0 0 0 0 0 ↵
Entrer la ligne 7 : 0 0 0 0 0 0 0 0 0 1 1 1 ↵

Voici l'image normale
-----
011111001110
000110001010
111111111111
000011100000
111111000000
000000000000
00000000111

Voici l'image compressée
-----
2 5 9 3
4 2 9 1 11 1
1 12
5 3
1 6

10 3

Voulez-vous quitter [O/N] ? O↵

```

```

*****
* Bienvenu sur le simulateur de Fax *
*****
Entrer le nombre de lignes L de votre image : 5 ↵
Entrer le nombre de colonnes C de votre image : 5 ↵

Saisie de l'image normale
-----
Entrer la ligne 1 : 0 0 0 0 0 ↵
Entrer la ligne 2 : 0 0 1 0 0 ↵
Entrer la ligne 3 : 0 1 1 2 0 ↵
Erreur de saisie. Ressaisir la ligne 3
Entrer la ligne 3 : ?

.
.
.

```