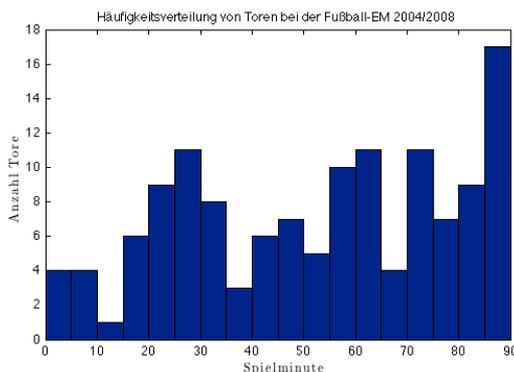


Problème – Histogramme

30 points

Un histogramme est un graphique obtenu en portant sur un axe les intervalles de classes d'une distribution statistique et, sur ces intervalles, des rectangles ayant une aire proportionnelle à l'effectif ou à la fréquence de la classe.

Par exemple l'histogramme ci-dessous montre le total de buts marqués pendant les matchs de foot du tournoi européen de foot en 2004 et 2008 réparti sur la durée d'un match. On voit clairement que le plus de buts (17) ont été marqués pendant les cinq dernières minutes d'un match.



Tâche

Écrivez un programme qui, pour chacune des **N** classes d'un histogramme, lit à partir du fichier HISTOGRAMME.TXT la valeur X_N de cette classe, et qui ensuite génère et écrit la représentation graphique de cet histogramme dans le fichier OUT.TXT.

Restrictions

$$1 \leq N \leq 100$$

Le nombre de classes

$$1 \leq X_i \leq 100$$

La valeur de la *i*-ième classe

Entrée et sortie du programme

Entrée

- La première ligne du fichier texte HISTOGRAMME.TXT contient un seul nombre entier positif: le nombre de classes **N** de l'histogramme.
- Les **N** lignes suivantes contiennent chacune un seul nombre entier positif: la valeur X_i de la *i*-ième classe.

Sortie

Le programme doit écrire dans le fichier OUT.TXT l'histogramme généré dans lequel chacune des **N** classes est représentée par une barre verticale dont la hauteur correspond à la valeur X_i de la classe. Chaque ligne du fichier OUT.TXT doit contenir exactement **N** caractères (1 caractère par classe) et seulement le caractère «#» et le caractère « » (un blanc) sont admissibles. Le nombre de lignes du fichier texte est variable et en fonction des valeurs X_i de l'histogramme. Le fichier OUT.TXT ne doit pas contenir de ligne vide, c.-à-d. la barre verticale la plus haute doit commencer dans la première ligne.

Exemples d'exécution

Exemple 1

Contenu du fichier HISTOGRAMME.TXT:

```
5
3
6
5
6
2
```

Contenu du fichier OUT.TXT:

```
# #
###
###
####
#####
#####
```

Exemple 2

Contenu du fichier HISTOGRAMME.TXT:

```
13
1
2
3
4
5
6
7
8
9
10
10
10
2
```

Contenu du fichier OUT.TXT:

```
###
####
#####
#####
#####
#####
#####
#####
#####
#####
#####
```

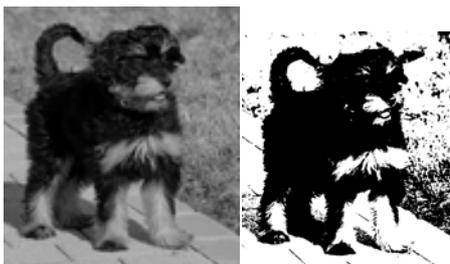
Remettez le programme sous le nom HISTOGRAMME.xxx, avec xxx=PAS ou C(PP). Remettez également le fichier binaire exécutable HISTOGRAMME.EXE correspondant au programme.

Problème – Seuil

30 points

Les logiciels de manipulation d'images et de photos (Gimp par exemple) offrent la possibilité de transformer une image initialement en couleurs ou en niveaux de gris en une image en noir et blanc purs. Dans cette image transformée, les pixels blancs représentent les pixels dont la valeur est strictement supérieure à un seuil fixé, et les pixels noirs représentent les pixels dont la valeur est inférieure ou égale à ce seuil. Le nom anglais de cet outil est «Threshold».

Exemple



Tâche

Écrivez un programme qui permet de créer une image en deux couleurs (noir et blanc) à partir d'une «image» en niveaux de gris composée de pixels dont les valeurs sont des entiers compris entre 0 (NOIR) et 127 (BLANC) inclus. Les dimensions de l'image constituent les deux premières lignes du fichier texte contenant l'image. Le nom de ce fichier texte doit être saisi au clavier.

Il faut ensuite déterminer la valeur seuil (médiane) qui permet de partager les valeurs des pixels en deux groupes, ceux dont la valeur est inférieure ou égale à la médiane deviendront noirs et ceux dont la valeur est supérieure à la médiane deviendront blancs.

Rappel

La médiane est la valeur qui permet de partager une série numérique ordonnée en deux parties de même nombre d'éléments

Restrictions

$$1 \leq L \leq 200$$

La largeur de l'image

$$1 \leq H \leq 200$$

La hauteur de l'image

Entrée et sortie du programme

Entrée

- La première ligne du fichier contient un seul nombre entier positif: la largeur **L** de l'image.
- La deuxième ligne du fichier contient un seul nombre entier positif: la hauteur **H** de l'image.
- Les **H** lignes suivantes contiennent chacune exactement **L** entiers compris entre 0 et 127 représentant les pixels formant l'image.

Sortie

Affichage à la console de la valeur de la médiane et création d'un fichier contenant l'image initiale transformée avec uniquement des pixels noirs et blancs. Le fichier porte le nom du fichier saisi préfixé par le mot «OUT». La première ligne du fichier résultat contient la largeur **L** de l'image alors que la deuxième ligne du fichier contient la hauteur **H** de l'image. Ces deux lignes sont suivies de **H** lignes contenant chacune **L** entiers ayant soit la valeur 0, soit la valeur 127.

Exemples d'exécution

Exemple 1

Dans cet exemple, le nom du fichier contenant l'image est SEUIL.TXT et est entré au clavier. Le programme affiche la valeur de la médiane, ici 14.

Le programme affiche donc:

14

Voici le contenu du fichier SEUIL.TXT (L = 10 et H = 20):

```

10
20
19 7 17 5 13 19 2 12 104 26
19 13 18 14 7 19 1 99 61 45
13 6 4 8 14 10 9 56 17 50
17 12 0 10 10 6 9 4 102 33
20 2 5 4 4 20 11 117 18 77
20 2 18 10 13 5 3 115 13 115
9 13 9 2 9 1 3 79 119 97
13 2 3 7 3 18 16 84 92 18
3 0 14 6 4 1 0 11 74 109
10 18 19 16 2 15 8 8 34 22
9 11 12 18 7 4 20 69 31 38
13 18 16 17 15 18 5 103 32 36
8 15 14 10 17 11 18 70 50 20
13 12 2 0 15 18 6 112 122 41
9 0 16 11 17 1 19 20 97 124
16 11 18 19 7 0 5 41 37 44
10 18 11 14 19 8 5 66 115 40
3 6 16 20 9 12 7 9 58 7
16 13 15 13 9 13 15 79 90 97
18 20 11 16 11 2 10 0 76 74
    
```

Le contenu du fichier résultat OUTSEUIL.TXT sera le suivant:

```

10
20
127 0 127 0 0 127 0 0 127 127
127 0 127 0 0 127 0 127 127 127
0 0 0 0 0 0 0 127 127 127
127 0 0 0 0 0 0 0 127 127
127 0 0 0 0 127 0 127 127 127
127 0 127 0 0 0 0 127 0 127
0 0 0 0 0 0 0 127 127 127
0 0 0 0 0 127 127 127 127 127
0 0 0 0 0 0 0 0 127 127
0 127 127 127 0 127 0 0 127 127
0 0 0 127 0 0 127 127 127 127
0 127 127 127 127 127 0 127 127 127
0 127 0 0 127 0 127 127 127 127
0 0 0 0 127 127 0 127 127 127
0 0 127 0 127 0 127 127 127 127
127 0 127 127 0 0 0 127 127 127
0 127 0 0 127 0 0 127 127 127
0 0 127 127 0 0 0 0 127 0
127 0 127 0 0 0 127 127 127 127
127 127 0 127 0 0 0 0 127 127
    
```

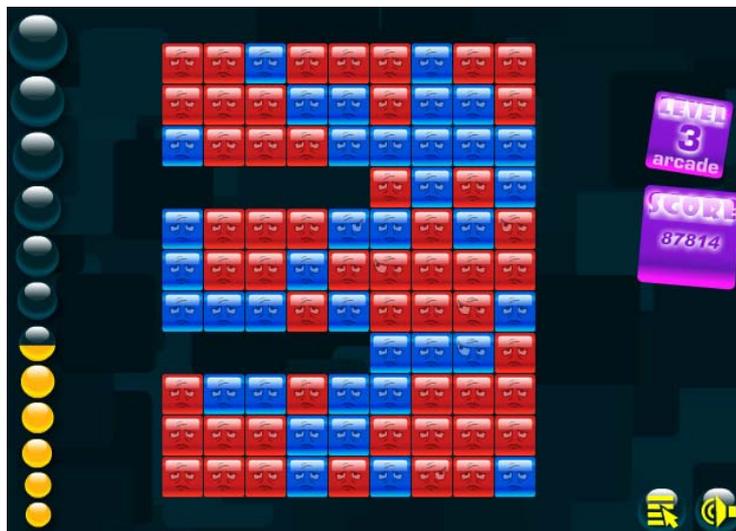
Le fichier SEUIL.TXT de cet exemple vous est fourni!

Remettez le programme sous le nom SEUIL.xxx, avec xxx=PAS ou C(PP). Remettez également le fichier binaire exécutable SEUIL.EXE correspondant au programme.

Problème – Colorgrid

40 points

Dans le jeu «Blocky» disponible sur internet, l'utilisateur est demandé de faire disparaître des blocs coloriés en choisissant quatre blocs de même couleur formant un rectangle, à l'intérieur duquel il ne peut y avoir de zone vide. De plus, le nombre de points attribué lors de chaque tour dépend de la taille de ce rectangle: plus le rectangle est grand, plus on gagne de points.



Tâche

Écrivez un programme qui lit à partir du fichier COLORGRID.TXT une plateforme de jeu, constituée de blocs coloriés et de zones vides, et qui détermine et affiche à la console le plus grand rectangle admissible que l'on peut choisir. Un rectangle est admissible si les quatre blocs coloriés formant le rectangle (les sommets du rectangle) ont la même couleur et si aucune zone vide ne se trouve à l'intérieur du rectangle. A part la couleur des quatre blocs coloriés formant le rectangle, la couleur des blocs à l'intérieur du rectangle n'a aucune importance.

Restrictions

$$1 \leq C \leq 100$$

Le nombre de colonnes de la plateforme du jeu

$$1 \leq L \leq 100$$

Le nombre de lignes de la plateforme du jeu

Entrée et sortie du programme

Entrée

- La première ligne du fichier texte COLORGRID.TXT contient un seul nombre entier positif: le nombre de colonnes **C** de la plateforme de jeu.
- La deuxième ligne du fichier texte contient un seul nombre entier positif: le nombre de lignes **L** de la plateforme de jeu
- Les **L** lignes suivantes contiennent chacune exactement **C** lettres majuscules qui représentent les blocs de la plateforme de jeu. La lettre majuscule «X» représente une zone vide, n'importe quelle autre lettre représente une autre couleur.

Sortie

Le programme doit afficher à la console une seule ligne contenant la coordonnée du bloc supérieur gauche du rectangle, la coordonnée du bloc inférieure droit du rectangle ainsi que la taille du rectangle. La coordonnée d'un bloc est formée du numéro de la colonne et du numéro de la ligne. La première colonne a le numéro 1 et la dernière colonne a le numéro **C** (les **C** colonnes sont donc numérotées de 1 à **C**). La première ligne a le numéro 1 et la dernière ligne a le numéro **L** (les **L** lignes sont donc numérotées de 1 à **L**). La ligne affichée doit respecter scrupuleusement le format des exemples d'exécutions.

Dans le cas où il n'est pas possible de trouver un rectangle, le programme doit afficher la phrase «Pas possible!».

Dans le cas où plusieurs solutions sont possibles, le programme peut librement choisir une de ces solutions.

Exemples d'exécution

Exemple 1

Contenu du fichier COLORGRID.TXT:

```
10
5
RGGGBBGGGR
GGBBBBBBGG
GBBBBBBBBG
GGBBBBBBGG
RGGGBBGGGR
```

Affichage à la console: (1,1),(10,5):50

Explication: Le plus grand rectangle est formé par les quatre blocs colorés en «R» qui se trouvent aux coordonnées suivantes: (1,1), (10,1), (1,5), (10,5). Ce rectangle comprend 50 blocs, c.-à-d. tous les blocs de la plateforme de jeu.

Exemple 2

Contenu du fichier COLORGRID.TXT:

```
10
5
RXXXYGGGR
GXXXBBBBGG
GBBBBBXXBG
GGBBBBXXGG
RGGGYGGGR
```

Affichage à la console: (5,1),(6,5):10

XSXSS

SSXS

Affichage à la console: Pas possible!

Exemple 3

Contenu du fichier COLORGRID.TXT:

```
5
4
SSXSS
XSSXS
```

Remettez le programme sous le nom COLORGRID.xxx, avec xxx=PAS ou C(PP). Remettez également le fichier binaire exécutable COLORGRID.EXE correspondant au programme.