

Épreuve pratique 2026 - Sujet 3 - Codage RLE d'images

Cette situation d'évaluation comporte ce document ainsi que des fichiers de codes et de données présents sur l'ordinateur à la disposition du candidat. Le candidat doit restituer ce document avant de sortir de la salle d'examen. Le candidat doit agir en autonomie et faire preuve d'initiative tout au long de l'épreuve.

En cas de difficulté, le candidat peut solliciter l'examinateur afin de lui permettre de continuer la tâche. Des moments privilégiés pour solliciter l'examinateur sont indiqués dans le document sous la forme d'appels professeur.

L'examinateur peut intervenir à tout moment, s'il le juge utile.

On considère dans ce sujet des images en niveaux de gris, c'est-à-dire composées de pixels décrits par une valeur entre 0 et 255 représentant l'intensité du niveau de gris, allant de noir pour la valeur 0 à blanc pour la valeur 255. Une image est alors vue comme une liste de valeurs entre 0 et 255 données par les valeurs des pixels ligne par ligne.

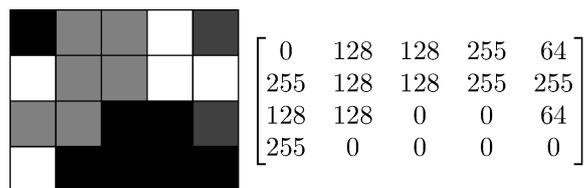


Figure 1. Exemple d'image en niveaux de gris et sa représentation en tableau de valeurs qui, une fois aplatie, devient $[0, 128, 128, 255, 64, 255, 128, 128, 255, 255, 128, 128, 0, 0, 64, 255, 0, 0, 0, 0]$ et de laquelle on peut retrouver l'image en connaissant sa largeur.

Les images manipulées sont des dessins ou des schémas présentant de grandes zones d'un même gris, l'objectif de ce problème est d'étudier une manière de les représenter efficacement en tirant parti de cette information. Pour cela, on remarque que les aplats font apparaître des valeurs qui se répètent dans les niveaux de gris des pixels, on va donc remplacer chaque suite de valeurs identiques par un couple (compte, valeur) indiquant le nombre de fois qu'une valeur est répétée ainsi que cette valeur. La liste de couples est elle-même aplatie, c'est-à-dire représentée par une liste de longueur paire $[\text{compte}_1, \text{valeur}_1, \text{compte}_2, \text{valeur}_2, \dots]$. On appelle codage RLE de l'image cette nouvelle liste, cela vient de l'anglais *run-length encoding*, car une suite de valeur identique est appelée *run*.

Exemple: La liste $[4, 4, 4, 0, 5, 5]$ présente trois fois de suite la valeur 4, ce qui correspond au couple $(3, 4)$, une fois la valeur 0 donc le couple $(1, 0)$ et enfin deux fois la valeur 5, donc le couple $(2, 5)$. La liste admet alors pour codage RLE la nouvelle liste $[3, 4, 1, 0, 2, 5]$. Ici, les deux listes sont de même longueur,

mais si la liste initiale était $[0, 0, 0, 0, 0]$ on aurait obtenu la liste $[5, 0]$ plus courte.

1. Déterminer si la liste obtenue par codage RLE est forcément de longueur inférieure ou égale à la liste de départ.

Appel 1	Appeler le professeur en cas de difficulté de compréhension du codage.
----------------	--

2. En étudiant bien la fonction `codage_rle` qui réalise le codage, écrire le corps de la fonction `decodage_rle` qui réalise le décodage d'une liste. Des tests sont fournis dans la fonction `test_codage()`, on pourra les compléter.

Appel 2	Appeler le professeur pour lui présenter votre fonction et son fonctionnement ou en cas de difficultés.
----------------	---

3. Pour tester le codage sur une image, on peut utiliser la fonction fournie `encoder_decoder_image` qui effectue le codage puis le décodage d'une image pour l'enregistrer à nouveau dans un fichier. Utiliser cette fonction sur les images `bac_nsi_32.png` et `bac_nsi_256.png` et observer la différence de comportement.
4. Le problème précédent est lié au fait que sur des grandes images, il est possible d'avoir plus de 255 pixels de la même couleur. Proposer une démarche de résolution de ce problème qui modifie les fonctions d'encodage et de décodage, puis l'implémenter.

Appel 3	Appeler le professeur pour lui présenter votre démarche ou en cas de difficultés.
----------------	---

Description du dossier

Le dossier fourni au candidat sur l'ordinateur comporte les éléments suivants :

- une version PDF de l'énoncé ;
- un code source de départ `rle.py` ;
- des images au format `png` en niveaux de gris auquel le sujet fait référence.

Préparation de l'environnement

Pour faire fonctionner le code fourni dans le dossier, les bibliothèques suivantes doivent être présentes : `pillow`.