

Problème #1: [20 pts]

Une image colorée a une largeur de 300 pixels et une hauteur de 200 pixels. L'image est codée en RVB où chaque composant est codé sur 8 bits.

1. Calculez le nombre des pixels dans cette image **[2 pts]**. Calculez la taille d'un pixel en bits **[2 pts]** et la taille de l'image en octets **[4 pts]**.
2. Si l'image est quantifiée en utilisant 3 bits pour la composante rouge, 3 bits pour la composante verte et 4 bits pour la composante bleue. Calculez la nouvelle taille de pixel et la nouvelle taille de l'image **[4 pts]**. Cette transformation est-elle avec ou sans perte, justifiez ! **[3pts]**.
3. L'image originale est sous-échantillonnée en prenant la moyenne de chaque 3x2 pixels. Calculez la nouvelle largeur et hauteur de l'image **[3 pts]**. En déduire la nouvelle taille de l'image en octets **[2 pts]**.

Problème #2: [10 pts]

Une image de 1200x800 pixels indexée contient 120 couleurs RVBA différentes.

1. **[4 pts]** Combien d'entrées y aurait-il dans la table d'index ? Calculez la taille de la table en octets.
2. Quelle sera la taille en octets de l'image:
 - a) **[4 pts]** on stocke la table d'index et l'image indexée
 - b) **[2 pts]** on stocke chaque pixel au format RVBA

Problème #3: [20 pts]

Une application produit des images dont la plupart des pixels sont noirs, blancs ou entièrement transparents, et quelques pixels sont colorés avec un certain niveau de transparence. L'image est stockée comme suit :

- 2 bits sont utilisés pour indiquer si le pixel est blanc, noir, transparent ou coloré (00 pour le noir, 11 pour le blanc, 01 pour une transparence totale et 10 pour un pixel coloré)
- Si le pixel est coloré alors 32 bits (8×4) qui représentent la valeur RVBA du pixel suivent les 2 bits précédents. En cas de blanc, noir ou entièrement transparent, nous ne stockons pas la valeur du pixel.

1. **[5 pts]** Comment va être stockée l'image RVBA suivante. (Utilisez 0, 1, 2, 3 à la place de 00 01 10 11)
(Note: A = 0 → entièrement transparent, A=255 non transparence)

(100,20,0,20)	(0,0,0,50)	(250,50,30,0)	(255,255,255,100)
(0,0,0,20)	(0,0,0,50)	(255,255,255,10)	(255,200,255,0)
(100,20,0,20)	(0,0,0,50)	(0,0,0,60)	(255,255,255,50)

2. **[10 pts]** Ecrivez un algorithme en pseudo-code pour transformer une image de RVBA en format décrit ci-dessus.
3. **[5 pts]** étant donné une image de 1000x2000 pixels dont 40%, 20%, 30%, 10% sont respectivement noirs, blancs, entièrement transparents et colorés. Calculez la taille de l'image en KOctets en utilisant le format RVBA complet et en utilisant le format décrit ci-dessus.

Problème #4: [10 pts]

Une source vidéo produit des images 300 x 200, entrelacées à une fréquence de champs (field) de 60 Hz, et en couleur (24 bpp). Vous convertissez d'abord la vidéo en représentation au format couleur 4:1:1. La composante Y du pixel est quantifiée à 8 bits et Cr, Cb à 6 bits chacun.

1. **[7 pts]** Calculez le taux de bits (bit rate) de la source vidéo.
2. **[3 pts]** Calculez la taille en octets d'une vidéo de 10 minutes.

Problème #5: [8 pts]

La matrice 4x4 ci-contre représente une image à 256 niveaux de gris. Convertissez-la en image binaire en appliquant les méthodes suivantes :

1. **[2 pts]** La conversion naïve
2. **[3 pts]** Le tramage à la moyenne (average dithering)
3. **[3 pts]** Le tramage à la médiane (median cut dithering)

100	181	240	80
130	30	90	110
60	40	5	95
65	80	70	80

Problème #6: [14 pts]

Un message contient deux symbols (A et B) avec une distribution de probabilités de 20% pour A et 80% pour B.

1. **[7 pts]** Encodez le message ABAA en utilisant l'encodage arithmetique (montrez toutes les étapes de calcul)
2. **[7 pts]** Décodezz le nombre 0.4 afin d'obtenir un message de longeur de 3 symboles (montrez toutes les étapes de calcul)

Problème #7: [18 pts]

Un message contient 6 symboles avec la distrubtion de probabilités ci-contre.

1. **[8 pts]** Appliquez le codage de Huffman afin d'obtenir les codes des symboles (montrez l'arbre d'encodage)
2. **[2 pts]** Calculez la moyenne de longueur des symboles.
3. **[4 pts]** Calculez l'entropie et l'efficacite de la compression.
4. **[2 pts]** Encodez le mot : ABED
5. **[2 pts]** Décodez le message: 111001111110

A	0.4
B	0.09
C	0.04
D	0.07
E	0.25
F	0.15

Bon Travail