

### **Problem #1: [20 pts]**

A given colored image has 400-pixel width and 300-pixel height. The image is coded in RGBA where each component is coded using 8-bits.

1. Compute the number of pixels in the image **[2 pts]**. Compute the pixel size in bits **[2 pts]** and the image size in bytes **[4 pts]**.
2. If the image is quantized by using 3 bits for the red component, 3 bits for the green component and 4 bits for the blue component and 1 bit for transparency. Compute the new pixel size and the new image size **[4 pts]**. Is this transformation loosely or lossless, justify! **[3pts]**.
3. The original image is sub-sampled by taking the average of each 4x4 pixels. Compute the new width and height of the image **[3 pts]**. Deduce the new image size in bytes **[2 pts]**.

### **Problem #2: [10 pts]**

An indexed image of 600x400 pixels have 480 different RGB full colors (8 bits per channel).

1. **[4 pts]** How many entries will be in the index table? Compute the size of the table in bytes.
2. What will be the size of the image in bytes if :
  - a. **[4 pts]** we store the index table and the indexed image
  - b. **[2 pts]** we store each pixel in RGB format

### **Problem #3: [16 pts]**

An application produce (24 bpp) RGB images where the most of pixels are monochromes, and few pixels are colored. A pixel is monochrome if the values of red, green and blue are equals.

The image is stored as following:

- Each pixel is stored in a distinct line. The line separator character is coded using 8 bits.
  - If the pixel is colored, we store the full color components (RGB).
  - If the pixel is monochrome, we store only one-component since R, G and B are equals.
1. **[3 pts]** How will be stored the following RGB, use NL as new line symbol.

(20,20,20)	(0,0,0)	(250,50,80)	(255,255,255)
(0,0,20)	(10,10,10)	(100,100,100)	(255,200,255)
(80,80,80)	(0,0,0)	(0,60,0)	(16,16,16)
  2. **[7 pts]** Write a pseudo-code algorithm to transform the above-described format to full RGB native format.
  3. **[6 pts]** Given an image of 1000x2000 pixels where the proportion of monochromes pixels is X.
    - a. Compute the image size in Kbytes using full RGB format (note: there is no new line symbol in this format)
    - b. Compute the image size in function of X using the above-described format.
    - c. Found the value of X such that the described format becomes better than (in term of storage size) the full RGB native format

### **Problem #4: [12 pts]**

A video source produces 400x200 images, interlaced at a frame rate of 25Hz, and full color (24 bpp). You first convert the video to 4:2:2 color format representation. The component Y of the pixel is quantized using 6 bits and Cr , Cb using 3 bits each.

1. **[8 pts]** Compute the bite rate of the video source.
2. **[4 pts]** Compute the size in bytes of a 10 minutes video.

## **Problem #5: [8 pts]**

Given the following message of 40 characters:

**ABBBBBCCCCMMMMMMXXWWWWWWKKKAAAAAAADDDED**

1. **[5 pts]** Encode the message using RLE encoding algorithm
2. **[3 pts]** Assume that each character and number uses 8 bits, compute the size of the message before and after encoding. Deduce the compression ratio.

## **Problem #6: [14 pts]**

A message contains three symbols (A, B and C). The LZW encoding algorithm starts with a dictionary that contains the three symbols A, B and C only with index 0, 1 and 2 respectively.

Construct the dictionary **[10 pts]** and give the outputted string **[4 pts]** for the following message:

**BACBABACBABA**CBBABAC****

## **Problem #7: [20 pts]**

A message contains 8 symbols with the following distribution probabilities

1. **[9 pts]** Apply the Huffman coding to obtain symbols' codes. Show the encoding tree, arrange the tree that the highest probability is at the right.
2. **[4 pts]** Compute the average symbol length
3. **[4 pts]** Compute the entropy and the compression efficiency
4. **[3 pts]** Encode the word : FAHED

A	0.2
B	0.05
C	0.01
D	0.08
E	0.15
F	0.02
G	0.09
H	0.4

---

*Good Work*