ECE 6354/5354 Fall 2016 HW 6 due 10/6

Problem 1 How many basis images are there in the 2-D DCT of a 4x2 image?

Problem 2. Design the max-lloyd **2-level** (L=2) optimal quantizer (t_1 t_2 t_3 r_1 and r_2) for digitizing samples of a time-v`arying analog signal whose voltage varies between 0 volt and 2 volts and is modeled as generating values according to the probability density function.

$$p_u(x) = \begin{cases} \frac{-x}{2} & for \quad -2.0 \le x \le 0.0 \\ 0 & else \end{cases}$$

Problem 3. Form an RGB image by decoding the BMP file listed below.

Byte#	1	2	3	4	5	6	7	8	9	10	11	12	13	14
value	66	77	126	0	0	0	0	0	0	0	118	0	0	0

Byte#	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34
value	40	0	0	0	4	0	0	0	2	0	0	0	1	0	4	0	2	0	0	0

Byte#	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54
value	8	0	0	0	0	0	0	0	0	0	0	0	16	0	0	0	16	0	0	0

Byte#	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74
value	0	0	0	0	255	0	0	0	0	255	0	0	0	0	255	0	1	1	1	0

Byte#	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94
value	12	12	12	0	23	23	23	0	34	34	34	0	45	45	45	0	56	56	56	0

Byte#	95	96	97	98	99	100	101	102	103	104	105	106	107	108	109	110	111	112	113	114
value	67	67	67	0	78	78	78	0	89	89	89	0	110	110	110	0	121	121	121	0

Byte#	115	116	117	118	119	120	121	122	123	124	125	126
value	132	132	132	0	4	1	0	0	4	128	0	0

Problem 4.
Encode the following 1-byte digital sequence using LZW encoding with 9-bits for all codewords (0-511 decimal) .
4, 3, 4, 3, 1, 3, 4, 3
Represent the encoded sequence using decimal values 0-511.

Problem 5
Given image $\{f(m,n)\} = \begin{bmatrix} [1] & 2 \\ 3 & 4 \end{bmatrix}$, compute the coefficients $\{F(u,v)\} = \begin{bmatrix} [F(0,0)] & F(1,0) \\ F(0,1) & F(1,1) \end{bmatrix}$ of the 2-D DCT transform (where we've used image indexing, not matrix indexing) and compute the four 2x2 basis images.

Problem 6.
a) Copy your file red_24bit_image.cpp to red_8bit_image.cpp.
b) Make red_8bit_image.cpp so that it writes out a BMP image file using 8-bits per pel and a colortable consisting of the red "colors" from 0 for black to 255 for brightest red.
Note:
You must modify the BITMAPINFOHEADER.
A 24-bit BMP image does not have a color table, but your 8-bit BMP files will have a colortable.
Therefore, in red_8b it_image.c, you will need to write out a color table at the appropriate place in the output file.
You will do this by mallocating a colortable vector containing 256 RGB quads.
In your colortable for "red colors", the RGBquad bytes for blue=0, green=0, and red will take values from 0 through 255.
Finally, you will write out one-byte per pel to the output file.
Steps:
1) Modify the BITMAPINFOHEADER to indicate an 8-bit BMP image.
2) Modify the rowsize to 4*int(width+3/4)
3) Make sure the imagedata is mallocated to be the correct size to accomodate 1-byte per pel
imagedata =(unsigned char *)malloc(rowsize*height*sizeof(unsigned char));

4) Mallocate an unsigned vector to hold the colortable RGB Quads
colortable =(unsigned char *)malloc(4*256*sizeof(unsigned char));
5) Assign the correct byte-value to each of the RGB values in the colortable RGBquads vector .
6) Make sure the filesize and offset to image data values in the BMP file header reflect the presence of the colortable and the use of 1-byte per pel.
7) Write out the colortable after the header (at the appropriate place in the output file).
8) Is there anything else you need to change in the program?

You will know that you have correctly taken care of every aspect of the 8-bit BMP image-file when your image opens up and displays correctly. E-mail me your 8-bit BMP image and c code file.