# Homework 1 Solutions

## CS414, Multimedia Systems (Instructor: Klara Nahrstedt)

**Posted: February 22, 2008**

**Due: February 29, 2008 at 11:59pm CST**

## Important Instructions

*This homework assignment should be done individually. Penalties for cheating as described in the grading policy on the course website apply.* Solutions should be done with a document preparation system, such as LaTeX or Microsoft Word (figures may be drawn by hand). In the homework solutions, you should show all of your work that you used to arrive at each answer to the problems. If possible, a hard copy of the assignment should be turned into the instructor at the beginning of class on Friday. Otherwise, a hard copy of the assignment should be slid under the door to the instructor's office (3104 SC) by the specified deadline.

## 1. Problem (4 Points)

Consider two types of data streams with the following characteristics:

- Stream 1: strongly periodic, weakly regular, discrete data stream;
- Stream 2: weakly periodic, strongly regular, discrete data stream;

Describe in detail a multimedia application, which uses both of these types of data streams.

## Answer:

The application that has the above characteristics is an MPEG Video movie application with advertisement clips and control commands.

1. MPEG video streams are strongly periodic and weakly regular, discrete data streams because the video frames are processed periodically, but the size of mpeg frames differs
2. Control commands, which accompany video streams such as pause, stop, and control the advertisement clips, generate weakly periodic, strongly regular, discrete data stream because the control commands are of the same size, but they are not sent in strongly periodic intervals. However, there might be some periodicity observed because the duration of movie segments does not differ too much from each other. The advertisement clips are usually of predefined length.

## 2. Problem (4 Points)

a. (2 Points) Explain Pulse Code Modulation (PCM). Specifically, what is PCM? How is PCM used on audio data?

**Answer:** PCM is the method of discrete sampling of continuous signal which uses the Nyquist result. It samples the signal at twice the desired highest frequency and (by Nyquist theorem loses no information upto that frequency.

b. (2 Points) Why is CD quality audio sampled at a 44.1 KHz frequency when the human hearing frequency range is only between 20Hz and 20 KHz?

**Answer:** Due to Nyquist's result, the sampling frequency has to be at least twice this figure to insure that no information is lost. So if 20 KHz is the max hearing frequency for humans, we need to sample at least at 40 KHz to ensure lossless information gathering. The standard then decided on 44.1 KHz

## 3. Problem (12 Points)

Consider the HDTV-progressive format used in the United States for video transmission between a television station and a receiver device. *(Note: This progressive format is 24 bits/pixel, 60 Hz frame rate, 1080 scan lines, 960 visible lines, and a 16:9 aspect ratio).*

a. (5 Points) Compute the bandwidth requirements for this format when only visible lines are transmitted.

**Answer:** Then number of pixels of one frame when all scanned lines are sent is:

(vertical pixels) * (horizontal pixels) = (1080 *16/9) * 1080 pixels

Number of pixels of one frame if visible lines are sent: (1080 *16/9)*960 = (1920 * 960) pixels

Assuming 24 bits per pixel and frame rate of 60fps, we get bandwidth (1920x960x24x60) bits per second

b. (5 Points) Compare the bandwidth requirement for the above format with the bandwidth requirement for an NTSC-interleaved TV format with 24 bits/pixel, 30 Hz frame rate, 525 scan lines, 486 visible lines, and a 4:3 aspect ratio. Assume that both streams are required to transmit all scanned lines.

**Answer:** Bandwidth for NTCS-i is: (525x4/3) x525x24x30 bits per second;

Bandwidth for HDTV-p is: 1920x1080x24x60 bits per second;

Bandwidth for HDTV-p is much higher than NTCS-i;

c. (2 points) Consider the time code format *HH*:*MM*:*SS*:*FF* defined for video by the Society of Motion Picture and Television Engineers (SMPTE) where *HH* denotes hours, *MM* denotes minutes, *SS* denotes seconds, and *FF* denotes frame number. Given the SMPTE timestamp 00:01:00:10, convert it back to the original frame number using the NTSC frame rate of 30 frames per second.

**Answer:** 1 minutes = 60 seconds. In 60 seconds we will see 60*30 = 1800 frames, plus we are at the frame 10, so the original frame number is 1800+10 = $1810^{th}$

## 4. Problem (10 Points)

Consider the following sequence of eighteen 8-bit audio samples: 245, 235, 236, 243, 257, 236, 243, 242, 245, 235, 236, 243, 257, 236, 243, 257, 236, 243. Notice that this sequence of eighteen samples takes 18*8=144 bits. Apply Differential Pulse Code Modulation (DPCM) to the sequence of samples. Also, do the following: (1) specify the coding table used, (2) specify the resulting DPCM-encoded sequence, and (3) compute how much compression can be achieved with DPCM.

**Answer:** The current sequence takes 144 bits. Let us create DPCM sequence:

245, 10, -1, -7, -14, 21, -7, 1, -3, 10, -1, -7, -14, 21, -7, -14, 21, -7

245, -10, 1, 7, 14, -21, 7, -1, 3, -10, 1, 7, 14, -21, 7, 14, -21, 7 (Also accepted)

We have now 8 different values in the sequence (245, 10, -1, -7, -14, 21, 1, -3), so we need only 3 bits to encode these three different values with assignments:

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 000 | 001 | 010 | 011 | 100 | 101 | 110 | 111 |
| 245 | 10 | -1 | -7 | -14 | 21 | 1 | -3 |
| 245 | -10 | 1 | 7 | 14 | -21 | -1 | 3 |

Then the sequence of the 18 samples is encoded as

000,001,010,011,100, 101, 011, 110, 111, 001, 010, 011, 100, 101, 011, 100, 101, 011

Number of bits needed is 3*18 = 54 bits, so the compression ratio is 54/144.

## 5. Problem (40 Points)

Consider the following image with an 8x8 pixels block.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 8 | 10 | 0 | 20 | 0 | 12 | 15 | 30 |
| 20 | 25 | 10 | 35 | 10 | 30 | 50 | 25 |
| 18 | 23 | 15 | 201 | 205 | 25 | 32 | 40 |
| 26 | 18 | 185 | 19 | 30 | 195 | 100 | 50 |
| 21 | 201 | 190 | 8 | 12 | 175 | 248 | 43 |
| 18 | 170 | 185 | 30 | 38 | 168 | 230 | 25 |
| 12 | 25 | 200 | 240 | 205 | 200 | 32 | 18 |
| 1 | 5 | 243 | 8 | 15 | 232 | 12 | 5 |

Perform the following operations on this block:

a. (20 Points) Compute the DCT coefficients (use the DCT formulas in the notes).

b. (10 Points) Quantize the transformed matrix (use the luminance quantization table in the notes for JPEG quantization).

c. (10 Points) Perform a zig-zag scan compressing the resulting sequence with run-length encoding.

**Answer:** The following solution shifts the DCT values from the range (0,255) to the range (-128,127) before performing operations on the block. However, full credit will also be given without the shift. Also, the following solution applies the floor function to the quantized coefficients before run-length encoding. Solutions using the ceiling function or rounding to the nearest integer will also be accepted.

a. DCT TRANSFORMATION

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 459.12 | -348.16 | -295.15 | -85.65 | -180.87 | -94.73 | -142.51 | -68.44 |
| -182.29 | -139.77 | -95.97 | -128.44 | -139.48 | -133.6 | -127.58 | -111.23 |
| -303.72 | -16.48 | -183.8 | -12.82 | -13.05 | -273.89 | -102.3 | -219.23 |
| -139.07 | -121.88 | -132.22 | -136.17 | -129.36 | -129.25 | -128.18 | -140.33 |
| -356.12 | 61.2 | -2.05 | -265.73 | -286.12 | 14.33 | -126.73 | -20.38 |
| -90.59 | -132.56 | -144.12 | -115.39 | -130.97 | -126.9 | -142.34 | -146.59 |
| -34.51 | -268.78 | -46.05 | -226.82 | -8.25 | -79.38 | -182.2 | -218.46 |
| -117.39 | -147.37 | -160.14 | -125.19 | -114.01 | -137.55 | -111.52 | -144.16 |

b. QUANTIZATION

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 28 | -30 | -22 | -7 | -11 | -4 | -3 | -1 |
| -17 | -12 | -8 | -8 | -7 | -4 | -2 | -2 |
| -31 | -2 | -12 | -1 | -1 | -5 | -2 | -3 |
| -9 | -7 | -6 | -5 | -3 | -3 | -2 | -2 |
| -15 | 2 | -1 | -6 | -5 | 0 | -2 | -1 |
| -3 | -3 | -3 | -2 | -2 | -2 | -2 | -2 |
| -1 | -5 | -1 | -3 | -1 | -1 | -2 | -3 |
| -2 | -3 | -3 | -3 | -2 | -2 | -2 | -2 |

c. RUN-LENGTH ENCODING (Special character is '!')

28 -30.0 -17.0 -31.0 -12.0 -22.0 -7.0 -8.0 -2.0 -9.0 -15.0 -7.0 -12.0 -8.0 -11.0 -4.0 -7.0 -1.0 -6.0 2.0 -3.0 -1.0 -3.0 -1.0 -5.0 -1.0 -4.0 -3.0 -1.0 -2.0 -5.0 -3.0 -6.0 -3.0 -5.0 -2.0 -3.0 -1.0 -2.0 -5.0 -3.0 -2.0!2 -3.0 -2.0 0.0 -2.0 -3.0!3 -1.0 -2.0!3 -1.0 -2.0 -1.0 -2.0!4 -3.0 -2.0!2

## 6. Problem (20 Points)

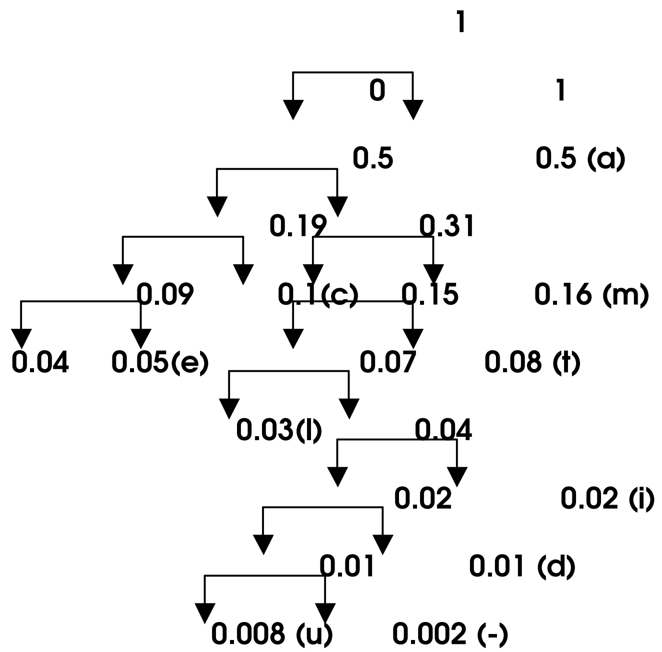
Consider the following alphabet {*a,c,d,e,i,l,m,s,t,u,-*} with the following probabilities:

> P(*a*) = 0.5, P(*c*)=0.1, P(*d*)=0.01, P(*e*)=0.05, P(*i*)=0.02, P(*l*) = 0.03, P(*m*)=0.16, P(*s*)=0.04, P(*t*)=0.08, P(*u*)=0.008, P(-)=0.002

a. (10 Points) Construct a Huffman tree and encode the word `*mat*'.

b. (10 Points) Encode the word `*mat*' with arithmetic encoding. Which coding is more efficient for this particular example, Huffman or Arithmetic? Why is one coding more efficient than the other? *(Note: you should show the conversion of the resulting value from arithmetic coding to its corresponding binary fraction representation to compare the efficiency).*

## Answer:




Huffman Code:

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 011 | 001 | 0000 | 0001 | 0101 | 01000 | 010011 | 0100101 | 01001000 | 01001001 |
| a | m | c | s | e | t | l | i | d | u | (-) |

Word `mat' is encoded as 011,1,0101 (need 8 bits to encode)

Arithmetic Coding (see separate file for complete solution): "mat" is a value between (0.5608,0.5672); Binary fractions: 0.1 : 0.5= $2^{-1}$, 0.01 : 0.25= $2^{-2}$, 0.001: 0.125=$2^{-3}$, 0.0001 :

0.0625=$2^{-4}$; Pick value 0.5+0.0625 = 0.5625 (is in the interval) = 0.1001 (need 4 bits to encode)

## 7. Problem (10 Points)

Consider a video-on-demand (VoD) system consisting of clients requesting video streams with the following characteristics: 640x480 pixels, 16 bits/pixel, and 25 fps. This VoD system runs on top of a transport layer protocol (similar to UDP) that uses 9-KByte packets. This system also uses admission control and runs a negotiation protocol. Provide the following information assuming that the available bandwidth at the VoD server at request time in the UDP-like transport level is 5 Mbps:

a. (5 Points) Specify the translation relation between the VoD application (in terms of video stream characteristics) and the UDP-like transport layer described above. Give the admission condition(s) at the transport layer at the VoD server. Also, determine how many clients the VoD server can support.

**Answer:** MA = 640X480x16 = 4,915,200 bits = 614,400 bytes = 600 KBytes

The bandwidth for one client-server stream is BN = 9 KB * $\lceil 600/9 \rceil$ * 25 = 15075 KB per second ~ 123 Mbps at the network subsystem level. The admission condition is $\sum BN_i < 5$ Mbps; zero streams satisfy this admission condition since a single stream requires at least 123 Mbps.

b. (5 Points) Specify the negotiation protocol between the VoD client and VoD service to setup this connection.

**Answer:** Since the client requests video stream quality, it will be receiver-initiated negotiation. The client gets the QoS parameters (MA, framerate) from the user, translates the parameters (to BN) and performs admission control at the transport layer. If the admission control is positive, reservation happens, the session layer negotiation service sends the application parameters (MA, framerate) parameters to the VOD server. The video server again performs translation to the transport system, admission control. If admission is positive, it reserves the bandwidth and sends the resulting application parameters back. In case of negative admission, the application quality parameters are degraded and changed parameters are sent to client. Once the client gets the resulting parameters, it compares the resulting parameters with the reserved/admitted parameters and adjusts accordingly.