

THIS PAPER IS NOT TO BE REMOVED FROM THE EXAMINATION HALLS

UNIVERSITY OF LONDON

291 0325 ZB

**BSc Examination**  
for External Students

**COMPUTING AND INFORMATION SYSTEMS AND  
CREATIVE COMPUTING**

**Data Compression**

Dateline: Friday 22 May 2009 : 2.30 – 4.45 pm

Duration: 2 hours 15 minutes

Candidates should answer **THREE** questions only. Full marks will be awarded for complete answers to **THREE** questions. There are 75 marks available on this paper.

A hand held calculator may be used when answering questions on this paper but it must not be pre-programmed or able to display graphics, texts or algebraic equations. The make and type of machine must be stated clearly on the front cover of the answer book.

**Question 1**

- (a) Consider a binary source  $S_1 = (A, B)$ , where  $A$  and  $B$  occur independently, and the extended alphabet  $S_2 = (AA, AB, BA, BB)$ . [10]
- (i) Prove that the (first-order) entropy of  $S_2$  is double the value of the entropy of  $S_1$
- (ii) Suppose that  $S_1 = (A, B) = (0.1, 0.9)$ . Demonstrate the entropy relationship between  $S_1$  and  $S_2$ . Show all your work.
- (b) Outline in a flowchart the adaptive Huffman algorithm for decoding. [5]
- (c) Demonstrate how your adaptive Huffman algorithm works for decoding 101000010101000001111010010001. Trace the values of the input, output, alphabet and the tree structure on each step. Write down the decoding result and compute the compression ratio. Assume that the fixed-length codes for characters  $A, \dots, Z$ , are 01000001,  $\dots$ , 01011010, respectively, and that each new symbol is added to the front (left most position) of the alphabet before a stable sort. [10]

## Question 2

- (a) What is a (binary) prefix code? Give an example of a prefix code and another example of a non-prefix code that is uniquely decodable. Explain your examples. [5]
- (b) Explain why there is no lossless compression program that can compress all the files even by one byte. Justify your answer. [5]
- (c) Explain the main ideas of *progressive image compression*. What are the main advantages of progressive image compression? Demonstrate, with the aid of an example using the sample image below, how a simple hierarchical (pyramid) coding works on both encoding and decoding ends. Assume a 3-layer pyramid structure and use the first elements as representatives. Show all your work. [15]

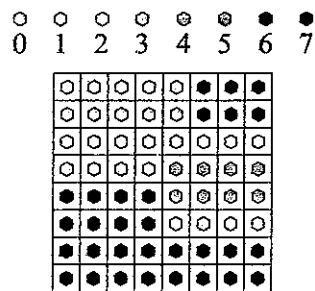
0	0	0	0	2	4	7	7
0	0	0	0	2	5	6	7
0	0	1	1	2	2	5	7
0	0	1	1	2	2	6	7
0	0	1	1	2	2	7	7
0	0	3	3	2	4	6	7
0	0	4	2	2	3	7	7
0	0	5	1	2	2	7	7

### Question 3

- (a) Derive step by step a canonical minimum-variance Huffman code for the message ABAABCCDEF. Is the Huffman code optimal for the source? If yes, explain why the Huffman code is optimal. Otherwise, describe a source for which the Huffman code would be optimal. Justify your answer. [10]

- (b) Describe the main ideas of the *predictive encoding* and *decoding*.

As an example, demonstrate how the image below can be compressed using the predictive coding. Explain how a predictive encoding may improve the compression ratio in terms of entropy change. Show all your work. [15]



#### Question 4

Consider the two  $8 \times 8$  matrices (a) and (b) below that represent two image sources respectively. The matrix entries represent the pixel colour values of the images.

(a)	0	0	0	0	2	4	7	7
	0	0	0	0	2	5	6	7
	0	0	1	1	2	2	5	7
	0	0	1	1	2	2	6	7
	0	0	1	1	2	2	7	7
	0	0	3	3	2	4	6	7
	0	0	4	2	2	3	7	7
	0	0	5	1	2	2	7	7
(b)	0	0	0	0	2	4	7	7
	0	0	0	0	2	5	6	7
	0	1	1	1	2	2	5	7
	1	1	1	1	3	4	6	7
	1	2	1	1	4	5	7	6
	2	2	3	3	4	4	6	6
	3	4	4	2	3	4	7	6
	5	5	5	6	5	7	6	5

Show all your work in answering the following questions.

- What is the minimum number of bits required for fixed-length coding for each of the two sources? [2]
- Draw a histogram (or bar-chart) to contrast the probability distribution of the two sources, plotting probability values against pixel colour values. [4]
- Which source would contain more information on average? Justify your answer. [4]
- Compress the two sources applying the canonical minimum-variance Huffman encoding algorithm directly. On which source does the Huffman algorithm achieve a better compression result? On which source does the Huffman algorithm achieve a performance that is closer to optimal? Justify your answers. [15]

### Question 5

- (a) Outline the Arithmetic decoding algorithm for a binary source in a flowchart.  
A binary sequence of length 4 (symbols) was encoded on the binary alphabet (B, W) using the Arithmetic encoding algorithm. Suppose that the probability  $Pr(B)$  is computed based on a previous input sequence BWWWBWWWW and the encoded output is 0.34.  
Demonstrate, with the aid of a diagram or a table, how the Arithmetic decoding algorithm derives the original sequence of symbols step by step. [10]
- (b) Explain why the Reflected Gray code is a good representation for coding the colours of greyscale images. Derive the *Reflected Gray code* for the decimal number 11. [5]
- (c) Explain briefly what is meant by a *cartoon-like image*. Give an example of such an image in real life. Provide a  $5 \times 5$  matrix of data to aid your explanation. [5]
- (d) Explain, with an example of data or diagram, the concept of *data rate* in the context of audio compression. What does the data rate measure? [5]