UNIVERSITY OF LONDON

291 0325 ZA

External Programme

B. Sc. Examination 2008

COMPUTER INFORMATION SYSTEMS

2910325 [Western] Data Compression

Duration: 2 hours and 15 minutes

Date and time: Tuesday 20 May 2008: 2.30 – 4.45 pm

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, text or algebraic equations. The make and type of machine must be stated clearly on the front cover of the answer book.

THIS EXAMINATION PAPER MUST NOT BE REMOVED FROM THE EXAMINATION ROOM

Question 1

(a) A compression process is often said to be 'negative' if its compression ratio value is greater than 1. Explain, with an example, why negative compression is inevitable for lossless compression in general. [5]

(b) Describe the main efficiency problem of the canonical minimum-variant Huffman coding algorithm that needs to maintain a frequency list. Explain and demonstrate, with an example, how the efficiency of the algorithm can be improved. [10]

(c) Encode a string `AABACCABBAAACCC` following the LZW algorithm. Assume that the dictionary initially contains single characters A-F and occupies cells at 0–5 only. Demonstrate the content changes of the main variables and the dictionary. [10]

Question 2

(a) Discuss the absolute limit of lossless compression by showing why more than 99% of files cannot be compressed even by one byte. [5]

(b) Demonstrate, with an example, how to improve the compression performance of the static Huffman algorithm on a small alphabet with an imbalanced probability distribution. [10]

(c) Explain the predictive rule of JPEG $x = (Q+S)/2$. Demonstrate, with a small example, how it can be applied in pre-processing. Assume the pixel layout [5]

T	S
Q	x?

(d) Consider part of a grayscale image with 16 shades of gray that is represented by the array A below:

```
0001 0010 1100 0110
0010 1000 1100 0110
0011 1100 1101 1011
```

Demonstrate how the image can be pre-processed by several bitplanes (bi-level images) and therefore may achieve a better compression ratio. [5]

Question 3

(a) Explain, with an example, the concept of *fidelity* in the context of audio compression. [5]

(b) Explain what an *optimal code* is in the context of data compression. Are Huffman codes optimal? Comment and justify, with an example, on the truth of the following statement: [10]

> "Huffman codes for text compression are optimal in general because probabilities of the characters can be a negative power of 2."

(c) Explain what is used to represent the so-called colour depth in a common RGB colour model. What is the value of the colour depth in a representation where 2 bytes are assigned to every pixel? If a total of 253 number distinct colours is required, what is the smallest coulour depth value required? Give your reasons. [5]

(d) Determine whether the following codes for the alphabet (A, B, C, D) are *uniquely decodable*. Justify your answer for each case. [5]

(i) (1, 10, 101, 0101)

(ii) (000, 001, 010, 111)

(iii) (0, 001, 10, 011)

(iv) (000, 010, 011, 1)

(v) (0, 01, 001, 0001)

Question 4

(a) Encode the following string using the HDC algorithm. Explain the meaning of each control symbol used. What is the compression factor? What is the entropy? [10]

␣␣␣␣␣K␣␣BB33221110KBCCCBC

(b) Explain why the Reflected Grey Code is a better representation than normal binary codes for coding the colours of greyscale images. Derive the *Reflected Grey Code* for the colour codes in decimal below. [5]

11	10
10	9

(c) Outline the adaptive Huffman coding and discuss the advantages of the approach over the static approach. Demonstrate how you would decode the following binary string by the adaptive Huffman decompression algorithm. Trace the states of the input, output, alphabet and the tree structure on each step. [10]

0100001010100000101010001111101001000100000.

Question 5

(a) Explain what is meant by a *minimum-variance Huffman code*. Demonstrate, with an example, what technique can be used to derive a minimum-variance Huffman code. You may focus on one step of the Huffman encoding algorithm. [5]

(b) Consider the task of sending a set of numbers (9, 11, 12, 13, 12, 15, 17, 19, 20, 22) over a mobile communication channel with as few number of bits as possible. Propose a coding scheme using residuals to achieve a good compression. Demonstrate all your compression and decompression work and evaluate your approach. Justify your choice of any standard compression methods. [10]

(c) Explain the predictive rule of JPEG $x = Q + (S - T)/2$. Demonstrate, with a small example, how it can be applied in pre-processing. Assume the pixel layout

T	S
Q	x?

[5]

(d) One important step of the Arithmetic decoding algorithm is to update boundary. Identify an assignment error in the Arithmetic algorithm below and correct the error: [5]

```
1. L <- 0 and d <- 1
2. If x is within [L,L+d*p1)
3.      then output s1, leave L unchanged, and
4.           set d<-d*p1
5.      else if x is within [L+d*p1, L+d)
6.           then output s2, set L<- L*d+p2 and d<-d*p2
7. If the_number_of_decoded_symbols
8.              < the_required_number_of_symbols
9.      then go to step 2.
```