THIS PAPER IS NOT TO BE REMOVED FROM THE EXAMINATION HALLS

UNIVERSITY OF LONDON **CO3325 ZB**

BSc Examination

COMPUTING AND INFORMATION SYSTEMS, CREATIVE COMPUTING AND COMBINED DEGREE SCHEME

Data Compression

Thursday 3 May 2018: 14.30 – 16.45

Time allowed: 2 hours and 15 minutes

There are **THREE** questions on this paper. Candidates should answer all **THREE** questions. All questions carry equal marks and full marks can be obtained for complete answers to **THREE** questions. The marks for each part of a question are indicated at the end of the part in [.] brackets.

There are 75 marks available on this paper.

A handheld 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.

Question 1

(a) Consider a binary code that is *not* a prefix code. Can we conclude that the code is therefore not uniquely decodable? Explain your answer. Use a code example of length 4 to support your answer. [5]

(b) What is the distinction between *lossy* and *lossless* data compression? What does lossy compression usually aim to do? Give an example of real-life data that would require lossless compression. [5]

(c) Explain why it is possible to derive two different Huffman codes for the same probability distribution of an alphabet. Briefly describe how Canonical and Minimum-variance Huffman coding addresses this. [5]

(d) Consider a binary source file consisting of characters A and B with probability of 0.2 for B. Discuss the cause of inefficiency of applying the static Huffman compression algorithm to this source, and compute how much less than optimal it is. Explain and demonstrate how the efficiency of encoding may be improved. [10]

Question 2

(a) Explain the major difference between the approaches of the Shannon-Fano and the static Huffman encoding algorithms. Demonstrate the difference by deriving a code for (A,B,C,D) with probability distribution (0.4, 0.4, 0.1, 0.1), applying the Shannon-Fano encoding and the static Huffman encoding algorithms individually. [5]

(b) What would be the output if the HDC algorithm is applied to the sequence below? Explain the meaning of each control symbol that you use. What is the compression saving (in percentage)? What is the entropy? [10]

`AAB␣␣␣␣␣␣T␣␣UUBAAAAB␣␣BBBAA`

The final numerical result for the entropy is not important and you may give an arithmetic expression only.

(c) Outline the Adaptive Huffman encoding algorithm. Demonstrate, with the aid of the input example CAAABB, how the Adaptive Huffman encoding algorithm works. [10]

In your answer trace step by step the progress of running the Adaptive Huffman encoding algorithm in terms of the input, output, alphabet and the tree structure using the initial input CAAABB.

Question 3

(a) Explain, with the aid of a small example in diagram, what *B picture* means and how it works in the context of video compression. [5]

(b) Describe the main idea of *predictive encoding*. Suppose the matrix below represents the pixel values (in decimal) of part of a grayscale image. Using the predictor $x = (Q + S)/2$ in JPEG $\begin{array}{|c|c|}\hline \text{T} & \text{S} \\ \hline \text{Q} & \text{x?} \\ \hline \end{array}$, illustrate step by step how the predictive encoding algorithm may be applied to the matrix: [10]

$$\begin{matrix} 1 & 1 & 1 & 1 \\ 5 & 1 & 1 & 1 \\ 5 & 5 & 5 & 5 \\ 7 & 9 & 5 & 5 \end{matrix}$$

(c) Following the approach of LZW algorithm, decode the tokens (`1, 1, 2, 1, 3, 3, 258, 259, 257, 261, 3`) step by step. Assume that the dictionary initially contains single characters A–Z and occupies cells at 1–256 only. Demonstrate the content changes of the main variables and the dictionary. [10]

END OF PAPER