UNIVERSITY OF LONDON

GOLDSMITHS COLLEGE

B.Sc. Examination 2014

Computing

IS53010A Data Compression

Duration: 2 hours and 15 minutes

Date and time:

There are five questions in this paper. You should answer no more than three questions. Full marks will be awarded for complete answers to a total of three questions. Each question carries 25 marks. 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.

Electronic calculators must not be programmed prior to the examination. Calculators which display graphics, text or algebraic equations are not allowed.

THIS PAPER MUST NOT BE REMOVED FROM THE EXAMINATION ROOM

Question 1

(a) Explain, with the aid of a small example, what is meant by *I picture* in the context of video compression. You may use diagrams to support your explanation. [5]

(b) Encode the string `AABACCABBAAACCC` following the LZW algorithm. Assume that the dictionary initially contains single characters A-F and occupies cells at addresses 0–5 only. Demonstrate in each step the content changes of the main variables x, $word + x$, the output and the dictionary. [8]

(c) A binary tree (0-1 tree) can be used to represent a variable length code. Consider each of the four codes below for the alphabet (A, B, C, D) and draw the binary tree for each code.

i. (0011, 0001, 110, 111)
ii. (110, 111, 0, 1)
iii. (0000, 001, 1, 0001)
iv. (0000, 0001, 001, 1)

For each tree drawn, comment on whether the code being represented by the binary tree is a prefix code, and justify your conclusion. [12]

Question 2

(a) What is the distinction between a *lossy* and *lossless* data compression? What does a lossy compression usually aim to do? Give an example of real life data for which a lossless compression would be suitable. [5]

(b) One important step of the Arithmetic decoding algorithm is to update boundaries. Explain, with the aid of a diagram, how the boundaries are defined and updated in the Arithmetic Coding Algorithm below. Identify an assignment error in the algorithm and correct the error. [10]

```
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.
```

(c) Following the approach of the LZ77 algorithm, demonstrate how to encode, step by step, the string `AABACCABBAAACCC`. Assume the length $H = 6$ for the history buffer and the length $L = 6$ for the lookahead buffer. [10]

Question 3

(a) A student has given the following answer as step (5) in her assignment of demonstrating the execution steps of the Adaptive Huffman encoding algorithm. Suppose the sequence of symbols to be encoded is `CAAABB` initially. Highlight your answer for step (5) in tracing the states (or values) of the `input`, `output`, `alphabet` and the `tree structure` on each step and give reasons if there is any difference between yours and her answer. Indicate if there is an error in her answer. [7]

```
(5)
read-input:   B
Output    :   h(DAG) ASCII("B")

A={A(3), C(1), B(1), DAG(0)}

Tree:       5
          /     \
       A(3)     2
                /    \
               1     C(1)
             /   \
          B(1)  DAG(0)
```

(b) Explain what is meant by a *minimum-variance Huffman code*. Demonstrate, with the aid of 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 and use the example of the alphabet (A, B, C, D, E) and its probability distribution (0.4, 0.2, 0.2, 0.1, 0.1). [7]

(c) Consider the binary source file consisting of characters A and B with probability of 0.2 for B. Discuss the typical measure of compression inefficiency when the static Huffman compression algorithm is applied to this source. Highlight the cause of the inefficiency and demonstrate how the efficiency may be improved. [11]

Question 4

(a) Consider a source file that consists of characters in the alphabet (A, B, C) with the probability distribution of (1/3, 1/3, 1/3). An 'expert' claims that the static Huffman coding algorithm can generate an optimal prefix code for compression of such a source. Do you agree with him? Give your reasons. [5]

(b) Derive the *Reflected Grey Code* (RGC) for each of the colour codes in decimal below. Explain why the RGCs are regarded as a better representation than normal binary codes for greyscale images. [5]

9	10
10	11

(c) David claims that the binary code (1, 01, 001, 010) is a prefix code since it satisfies the Kraft inequality. Check if the code indeed satisfies the Kraft inequality and explain what is wrong with David's claim. [6]

(d) Decode the compressed string below using the HDC algorithm. Explain the meaning of each control symbol used. What is the compression ratio? What is the entropy of the source? [9]

```
r4n1Ar2n6BB3322r31n30ABr3Cn2BC
```

Question 5

Consider the two 8×8 matrices (i) and (ii) below that represent two independent image sources. The matrix entries represent the pixel colour values of the images. Show all of your work in answering the questions below.

(i)

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

(ii)

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

(a) What is the minimum number of bits required for the fixed-length coding for each of the two sources? Justify your answer. [3]

(b) Draw a histogram to contrast the frequency distribution of the two sources plotting frequency values against pixel colour values. [4]

(c) Outline, in plain English, example or flowchart, an algorithm that can determine which of the two sources contains more information on average. [5]

(d) Compress the two sources applying the canonical minimum-variance Huffman encoding algorithm directly. [6]

(e) Demonstrate how a simple differential compression scheme on rows may be applied as a preprocessing step before the Huffman algorithm. Explain how to estimate on which of the two sources your preprocessing scheme may achieve a better compression result. [7]