University of Glasgow

Friday 11 December 2020
1.00 pm – 4.00 pm
(3 hour)

DEGREES of MSci, MEng, BEng, BSc, MA and MA (Social Sciences)

ALGORITHMICS I (H)

Answer all 4 questions

This examination paper is worth a total of 60 marks.

1. **(a)** In at most **100 words** define what is meant by a border of a string. Give an example of a string of length 6 whose longest border is of length 4.

[3]

(b) Using your own words and in at most **300 words** describe the contents of the border table that is used in the Knuth-Morris-Pratt (KMP) string searching algorithm. Explain briefly how this table is used to determine the appropriate action when a mismatch is detected between the ith character of the text and the jth character of the string or pattern being searched for.

[7]

(c) Construct the border table of the KMP algorithm when the string being searched for equals:

$$ACGACACGTACGAA$$

where the text is over the alphabet $\Sigma = \{A, C, G, T\}$. [5]

2. Recall given a set of character frequencies a corresponding Huffman tree is constructed as follows:

- first add the leaves of Huffman tree which are characters with their frequencies labelling the nodes;
- second, while there is more than one parentless node, add a new parent to nodes of smallest weight, where the weight of the new parent node equals the sum of the weights of the child nodes.

The table below includes the number of occurrences of each character in the text:

"a simple string to be encoded using a minimal number of bits"

a	*b*	*c*	*d*	*e*	*f*	*g*	*i*	*l*	*m*	*n*	*o*	*p*	*r*	*s*	*t*	*u*	*space*
3	*3*	*1*	*2*	*5*	*1*	*2*	*6*	*2*	*4*	*5*	*3*	*1*	*2*	*4*	*3*	*2*	*11*

(a) Draw a Huffman tree for this text and write the subsequent code words for the characters '*f*' and '*i*'.

[8]

(b) Explain in your own words and using no more that **300 words** how, given a set of character frequencies, a corresponding Huffman tree can be constructed with a complexity of $O(m \log m)$ where m is the number of characters appearing in the text. You do not need to describe in any detail standard operations based on any underlying data structure that you might use.

[7]

3. **(a)** Explain what it means for a problem to be NP-complete and then discuss in your own words (using at most **150 words**) the implications, from the algorithmic point of view, of proving that a decision problem is NP-complete.

[5]

(b) Consider the following problems:

- **Graph colouring (GC)**
  Instance: a graph G and a target integer k.
  Problem: can one of k colours be attached to each vertex of G so that adjacent vertices always have different colours?
- **Workshop Programming (WP)**
  Instance: given a set of workshops W and k dates, workshop attendees A, a mapping $f : W \to 2^A$ such that $f(w)$ returns the set of attendees of the workshop w and an integer k.
  Problem: Is there a workshop progamme over the k dates such that every attendee has at most one workshop on any date?

Under the assumption that **GC** is NP-complete, prove that **WP** is also NP-complete.

Hint: consider when each attendee attends exactly two workshops.

[10]

4. **(a)** Describe, using a diagram or otherwise, a deterministic finite state automaton to recognize the language L consisting of all strings over the alphabet $\{a, b\}$ that contain two consecutive a's or two consecutive b's.

[4]

(b) Give a regular expression for the language L.

[2]

(c) Describe, using a suitable form of pseudocode, or otherwise, a Turing Machine that recognizes the language consisting of all strings over the alphabet $\{a, b\}$ that contain more a's than b's.

[9]

END OF QUESTION PAPER