The Open University of Sri Lanka
Faculty of Engineering Technology
Department of Electrical and Computer Engineering

General Instructions

1. Read all instructions carefully before answering the questions.

2. This question paper consists of **seven (7)** questions in **four (4)** pages.

3. Answer any **FIVE (5)** questions only. All questions carry equal marks.

4. Answer for each question should commence from a new page.

5. Answers should be in clear handwriting and do not use Red colour pen.

6. Clearly state your assumptions, if any.

7. This is a **Closed Book Test** (CBT).

Q1

(a) Briefly explain why the *Time complexity* is significant in algorithms.　　　　[02]

(b) What are the Asymptotic notations and explain them briefly.　　　　[03]

(c) Each of the expressions below gives the processing time spent by an algorithm for solving a problem of size n. By explaining your work find the lowest Big-Oh complexity of each problem.　　　　[15]

i)　　$n^2 \log_2 n + n(\log_2 n)^2$

ii)　　$3 \log_8 n + \log_2(\log_2(\log_2 n))$

iii)　　$n \log_2 n + n(\log_2 n)^2$

iv)

```
For (int i = n; i > 0; i /= 2) {
    For (int j = 1; j < n; j *= 2) {
        For (int k = 0; k < n; k += 2) {
            ... // constant number of operations
        }
    }
}
```

v)

```
For (i = 1; i < n; i *= 2) {
    For (j = n; j > 0; j /= 2) {
        For (k = j; k < n; k += 2) {
            sum += (i + j * k);
        }
    }
}
```

Q2

(a) Write the pseudocode algorithms for the following standard operations in stack data structure. State any assumptions you make.

i)　*initialize()*　　　　ii)　*isEmpty()*　　　　iii)　*isFull()*

iv)　*pop()*　　　　v)　*push()*

　　　　[10]

(b) Convert the following algebraic expression to *postfix* form using a stack as in the table below. Show the steps for each character:

$$[(A + B) \times (C + D)^\wedge(E + F)]/ G$$

| Reading character	Postfix	Stack

　　　　[10]

Q3

    (a) Describe what is a binary search tree (BST)? [02]

    (b) Construct a BST for the following sequence of numbers by taking the first element as the root: [12]

$$\{13, 3, 4, 12, 14, 10, 1, 8, 2, 7, 9, 11, 6, 18, 15\}$$

    (c) Use the BST in (b) to write all three traversals. [03]

    (d) Draw the new trees after deleting:

        i)   The element 14 from the tree drawn in (b), [01]

        ii)   The element 13 from the tree drawn in (d) - i). [02]

Q4

Consider the following directed, weighted graph to answer the questions below.
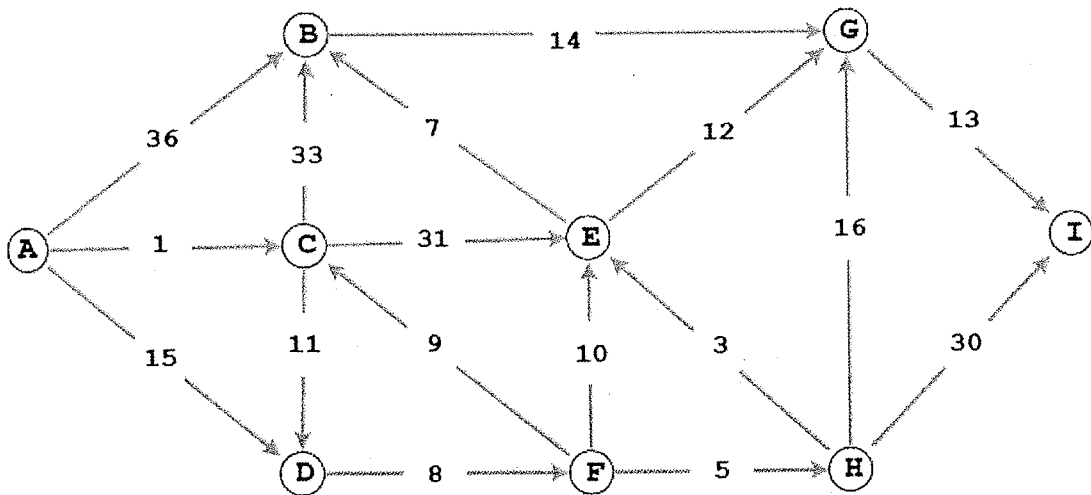
    (a) Step through Dijkstra's algorithm to calculate supposedly shortest paths from **A** to every other vertex. <u>Show your steps in the table form</u>. Cross out old values and write in new ones, from left to right, as the algorithm proceeds. [16]

    (b) Obtain an adjacency matrix for the above graph. [02]

    (c) Show the above matrix as an adjacency list. [02]

Q5

Consider the following operands and operators with their precedence for logic expressions. Note that you must consider left and right parenthesis as well.

Operands	Operators and their meaning		Precedence
0 1	~ & |	not and or	First Second Third

(a) Write the pseudocode algorithm to convert a string of logic expression into its expression tree. You can use any data structure to represent the tree and should explain how your data structure logically relates to the tree. Since ~ is a unary operator, you should put its only operand to its right child. [12]

(b) Draw the expression tree of the following logical expression: [04]

$$\sim (0 \ \& \sim 1 \ \& \ 0 \ | \ 0) \ \& \ 0$$

(c) Obtain the prefix and postfix notations for the given expression in (b). [04]

Q6

(a) What is hash collision? [03]

(b) Describe three hash collision resolution techniques? [03]

(c) Use one of the techniques you mentioned in (b) to construct a non-collision hash table of length 13 for the given keys in the same order,

{4684, 4879, 5651, 1829, 1082, 7107, 1628, 2438, 3951, 4758, 6967, 4989}

by using the hash function,

$h(k)$ = (sum of all digits of a key) MOD 13. [12]

(d) Give an advantage and disadvantage of each of the technique that you used in (c). [02]

Q7

(a) Briefly explain with illustrations, the workings of:

   i)   Selection sort algorithm; [05]

   ii)   Quicksort algorithm; [05]

(b) Using the last item as the pivot in Quicksort, explain with illustrations, why Quicksort performs very poorly (in terms of runtime) if the items are already in sorted order. You can use any sorted array with at least eight (8) numbers. [10]

-- End –