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

501

Study Programme	: Bachelor of Software Engineering Honours
Name of the Examination	: Final Examination
Course Code and Title	**: EEX4465 – Data Structures and Algorithms**
Academic Year	: 2023/2024
Date	: 11th August 2024
Time	: 1330-1630hrs
Duration	: **3 hours**

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. This is a Closed Book Test **(CBT).**
6. Answers should be in clear handwriting.
7. Do not use **red** colour pen.

Question 01 (20 Marks)

(a) Briefly explain why the Time complexity is significant in algorithms. [03]

(b) A logarithmic algorithm with processing time $T(n) = c \log(n)$ spends $T(N)$ seconds for processing N items, and c is the constant. How much time will be spent for processing $n=10000$ items, when $N= 500$ and $T(N) = 2ms$? [05]

(c) The following expressions give the processing time spent by an algorithm for solving a problem of size n. By explaining your work, find the worst-case complexity (Big-O) of each problem. [12]

a.

```
public static int F(int n) {
    if (n <= 1) return 1;
    return F(n-1) + F(n-2);
}
```

b.

```
public static int G(int n) {
    if (n <= 1) return 1;
    return 2 * G(n/2);
}
```

c.

```
public static int H(int n) {
  if (n <= 1) return 1;
  return H(n-1) + n;
}
```

d.

```
public static int I(int n) {
  int sum = 0;
  for (int i = 0; i < n; i++) {
    for (int j = 0; j < n; j++) {
      sum += i * j;
    }
  }
  return sum;
}
```

Question 02 (20 Marks)

(a) Give one advantage and one disadvantage of the linear probing technique. [02]

(b) Insert the objects with key values {15, 17, 8, 23, 3, 5}, respectively, into an initially empty hash table of size 7 with the hash function h(x) = x MOD 7, by using the following methods. Show the resulting table for each method.

i. Open addressing with linear probing [03]

ii. Closed addressing with direct chaining [03]

(c) Consider an implementation of a hash table with linear probing as given below

> It uses an array, "storage" to keep pointers to the keys, and uses a Boolean array, "used" to record which locations in "storage" have contained a "key"

i. Write the function, "lookup(. . .)" in pseudocode for this hash table implementation, searching for a given "key", and returning (as a Boolean) whether it was found. [12]

Question 03 (20 Marks)

(a) Consider the infix notation given by

$$4 \times (3 + 2/1) - 5$$

i. Write the above expression into its equivalent postfix notation. [01]

ii. Evaluate the postfix notation obtained in i) by showing the contents of the stack at every step, by creating the table with the column headers as shown below. [05]

Next symbol	Operand 1	Operand 2	Value	Stack

(b) Convert the following algebraic expression to postfix form using a stack as in the table below. Show the steps for each character: [14]

$$P/(Q - R) + S \times T$$

Reading character	Postfix	Stack

Question 04 (20 Marks)

(a) Write pseudocode algorithms for the operations below in a Binary Search Tree (BST):

i. To delete a node from a BST using a recursive function. [4]
ii. To search for a given value in a constructed BST using a recursive function. [4]

(b) Consider the sequence of numbers given below.

{20, 8, 22, 4, 12, 10, 14}

i. Construct the BST for the above sequence by taking the first element as the root. [4]
ii. Obtain the in-order traversal of the BST drawn in part i. [4]
iii. Re-draw the BST after inserting the new element 13 to the tree drawn in part i. [4]

Question 05 (20 Marks)

Given a map of cities and roads represented as a graph *G*, develop a plan to find a route that visits each city at least once. Use the Breadth-First Search (BFS) algorithm to ensure all cities are visited in the shortest path order from the starting city ***S***.

Input: A graph G represented as an adjacency list and a starting node ***S***.
Output: The order of cities visited.

(a) Write the pseudocode for the BFS algorithm applied to this scenario. Make sure your pseudocode clearly describes the process of exploring each city, marking it as visited, and exploring its adjacent, unvisited cities. [8]
(b) Given the graph G and the starting node ***S***, draw the BFS tree that shows the order in which the cities are visited. Indicate the levels of the tree and clearly mark the parent-child relationships between the cities. [8]
(c) Analyze the time and space complexity of the BFS algorithm in terms of the number of vertices V and edges E in the graph G. Provide a detailed explanation of your analysis. [4]

Question 06 (20 Marks)

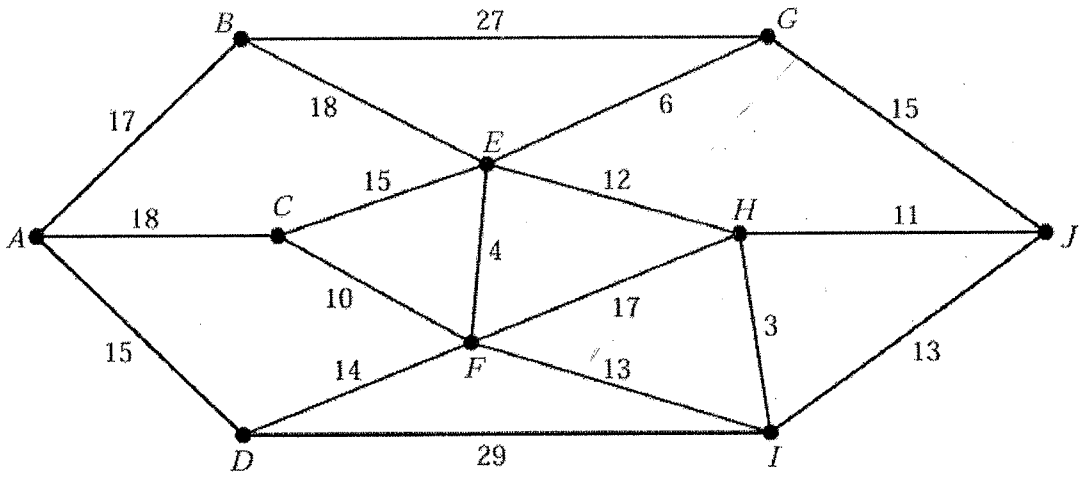

Figure 1: Network of roads

Figure 1 shows a network of roads. The number on each arc represents the length of that road in kilometers (*km*).

(a) Use Dijkstra's algorithm to find the shortest route from A to J.
   i. State your shortest route and its length.
   ii. Explain how you determined the shortest route from your labeled diagram. [8]

(b) Explain how you determined the shortest route from your labeled diagram. [4]

(c) The road from C to F will be blocked. Find the shortest route from A to J that does not path from C to F and state its length. [8]

Question 07 (20 Marks)

Merge sort follows the rule of Divide and Conquer to sort a given set of numbers/elements recursively. Answer the following:

(a) Briefly describe the rule of Divide and Conquer. [4]

(b) Write the pseudocode algorithm for the Merge sort. [8]

(c) Use the Merge sort algorithm to sort the following sequence of numbers. Show the complete step-by-step procedure: [8]

{18, 5, 21, 3, 13, 8, 2}

-END-