18

(Knowledge for Development)

KIBABII UNIVERSITY

UNIVERSITY EXAMINATIONS
2022/2023 ACADEMIC YEAR

END OF SEMESTER EXAMINATIONS
SPECIAL/SUPPLEMENTARY

FOR THE DEGREE OF
BACHELOR OF SCIENCE
(COMPUTER SCIENCE)

COURSE CODE : CSC 225

COURSE TITLE : DATA STRUCTURES

DATE: 10/08/2023 **TIME: 8AM-10AM**

INSTRUCTIONS TO CANDIDATES

ANSWER **QUESTIONS ONE** AND **ANY OTHER TWO.**

QUESTION 1 (COMPULSORY) **[30 MARKS]**

a) If you have to solve the searching problem for a list of n numbers, how can you take advantage of the fact that the list is known to be sorted? Give separate answers for

   i. lists represented as arrays. **[1 mark]**

   ii. lists represented as linked lists. **[1 mark]**

b) Show the stack after each operation of the following sequence that starts with the empty stack:

`push(a), push(b), pop, push(c), push(d), pop` **[1mark]**

c) Show the queue after each operation of the following sequence that starts with the empty queue:

`enqueue(a), enqueue(b), dequeue, enqueue(c), enqueue(d), dequeue`

[1mark]

d) Why is sorting necessary? **[1 marks]**

e) Consider the binary tree below:

```
        14
       /  \
      2    11
     / \   / \
    1  3  10  30
          /  /
         7  40
```

Write the order of the nodes visited in: **[6 marks]**

i. An in-order traversal,

ii. A pre-order traversal,

iii. A post-order traversal:

f) Given an array containing the digits 71808294, show how the order of the digits changes during each step of quicksort (using the array-based quicksort). Show the array after each swap. [3 marks]

g) In an array based implementation of the queue data structure, what strategy is used to avoid an array out of bounds error when the `int` variable, `rear,` storing the index of the next available array cell is incremented when its current value is N-1 where N is array size?

[2 marks]

h) In recursive methods, we have the base case and the inductive/recursive case. What is the importance of the base case? **[2 marks]**

i) Draw a complete binary tree with exactly six nodes. Put a different value in each node. Then draw an array with six components and show where each of the six node values would be placed in the array (using the usual array representation of a complete binary tree). **[5 marks]**

j) What are the steps to inserting a new item at the head of a linked list? [3 marks]

k) Outline any two applications of the stack data structure **[2 marks]**

l) Briefly describe the following data structures. **[4 marks]**

   i. Stack

   ii. Queue

   iii. Linked list

   iv. Binary tree

QUESTION 2 (20 marks)

a) What is the function of the reference `head/first` when used with a linked list? What is the data type of the head variable? **[2 marks]**

b) Draw a diagram of a linked list that contains nodes with data items of type String that contains the name of a city and type double that contains a pollution index. Include an instance variable named head to indicate the beginning of the list. Insert the following nodes: Franklin, 15.7, Chicago, 23.2, Denver, 7.2. **[3 marks]**

c) Create a generic Node class to represent the linked list depicted in your diagrams above. **[10 marks]**

d) Write a method called displayList that displays the data items in the Node class created in number `(c)` above. **[5 marks]**

QUESTION 3 (20 marks)

a) Give two different reasons to explain why the following binary tree is not a heap: **[2 marks]**

```
          91
         /  \
      77      46
     /  \       \
   68    81      11
```

b) Draw a new heap that is created by inserting 82 into the following heap: **[2 marks]**

```
          910
         /  \
      77      66
     /  \    /  \
   68    1  3    11
```

c) What problem does binary search tree suffer from? **[2 marks]**

d) Describe any two methods for storing binary trees in the computer. **[2 marks]**

e) Suppose characters `a, b, c, d, e, f` have probabilities `0.07, 0.09, 0.12, 0.22, 0.23, 0.27`, respectively.

   i. Find an optimal Huffman code and draw the Huffman tree. **[8 marks]**

   ii. What is the average code length? **[4 marks]**

Question 4

a) Describe why a very large hash table will likely increase the performance (i.e. faster additions and lookup) at the expense of wasting memory, and vice versa, why a small hash table will use less memory but result in a decrease in performance. **[4 marks]**

b) Suppose that an open-address hash table has a capacity of 811 and it contains 81 elements. What is the table's load factor? **[2 marks]**

c) Define the following as relates to hash tables:

   i. Collision [1mark]

   ii. Perfect hashing function [1 mark]

   iii. Load factor [2 marks]

d) Briefly describe one algorithm that is used for resolving collisions in a hash table. [4 marks]

e) Draw a hash table with open addressing and a size of 11. Use the hash function "k%11". Insert the keys: 5, 29, 20, 0, 27 and 18 into your table (in that order).[6 marks]

Question 5

a) Implement the following method. Do not use any local variables or loops. **[7 Marks]**

```
    public static void pattern(int n)

// Precondition: n > 0;
 // Postcondition: The output consists of lines of integers. The
//first line is the number n. The next line is the number 2n.
The next //line is the number 4n, and so on until you reach a
number that is //larger than 4242. This list of numbers is then
repeated backward //until you get back to n.

    /* Example output with n = 840:

    840
    1680
    3360
    6720
    6720
    3360
    1680
    840

*/
```

b) Write a recursive method that has one parameter which is an int value called x. The method prints x asterisks, followed by x exclamation points. Do NOT use any loops. Do NOT use any variables other than x. **[6 marks]**

c) Suppose that p, q, and r are all references to nodes in a linked list with 15 nodes. The variable p refers to the first node, q refers to the 8th node, and r refers to the last node. Write a few lines of code that will make a new copy of the list. You code should set THREE new variables called x, y, and z so that: x refers to the first node of the copy, y refers to the 8th node of the copy, and z refers to the last node of the copy. **[7marks]**