



The University of Fiji

School of Science and Technology



Department of Computer Science/ Information Technology

ITC215

Data Structures and Algorithms

Semester I, 2014

FINAL EXAMINATION QUESTION PAPER

Time Allowed: 3 Hours [10 Minutes Extra Allowed For Reading the Paper]
Weighting: 50%
Total Marks: 100 marks.

Part I:

Multiple Choice - 20 marks

Part II:

Short Answer Question 1 - 10 marks
Short Answer Question 2 - 15 marks
Short Answer Question 3 - 10 marks
Short Answer Question 4 - 10 marks
Short Answer Question 5 - 10 marks

Part III:

Long Answer Question - 25 marks

Total 100 marks

Instructions

- Show your answers clearly in each question to gain maximum marks.
- You may ask for extra sheets from the supervisors if the need arises.
- Do not forget to write your name and ID number on all the extra sheet(s) attached.
- Programmable calculators are not allowed.
- The paper contains 6 pages.

Section A: Multiple Choice

[20 Marks]

1. The memory address of the first element of an array is called
 - (a) floor address
 - (b) foundation address
 - (c) first address
 - (d) base address
2. Which of the following is not the required condition for binary search algorithm?
 - (a) The list must be sorted
 - (b) there should be the direct access to the middle element in any sublist
 - (c) There must be mechanism to delete and/or insert elements in list
 - (d) none of above
3. Which of the following is not a limitation of binary search algorithm?
 - (a) must use a sorted array
 - (b) requirement of sorted array is expensive when a lot of insertion and deletions are needed
 - (c) there must be a mechanism to access middle element directly
 - (d) binary search algorithm is not efficient when the data elements are more than 1000.
4. Which of the following data structure can't store the non-homogeneous data elements?
 - (a) Arrays
 - (b) Records
 - (c) Pointers
 - (d) None
5. Which of the following data structure store the homogeneous data elements?
 - (a) Arrays
 - (b) Records
 - (c) Pointers
 - (d) None
6. Which of the following name does not relate to stacks?
 - (a) FIFO lists
 - (b) LIFO list
 - (c) Piles
 - (d) Push-down lists
7. The term "push" and "pop" is related to the
 - (a) Array
 - (b) Lists
 - (c) Stacks
 - (d) all of above
8. When inorder traversing a tree resulted E A C K F H D B G; the preorder traversal would return
 - (a) FAEKCDHBG
 - (b) FAEKCDHGB
 - (c) EAFKHDCBG
 - (d) FEAKDCHBG
9. The complexity of linear search algorithm is
 - (a) $O(n)$
 - (b) $O(\log n)$
 - (c) $O(n^2)$
 - (d) $O(n \log n)$
10. The operation of processing each element in the list is known as
 - (a) Sorting
 - (b) Merging
 - (c) Inserting
 - (d) Traversal
11. The goal of hashing is to produce a search that takes
 - (a) $O(1)$ time
 - (b) $O(n^2)$ time
 - (c) $O(\log n)$ time
 - (d) $O(n \log n)$ time

12. A full binary tree with n leaves contains
 (a) n nodes. (b) $\log n$ nodes.
 (c) $2n - 1$ nodes. (d) n^2 nodes.
13. A graph with n vertices will definitely have a parallel edge or self loop if the total number of edges are
 (a) more than n
 (b) more than $n + 1$
 (c) more than $(n + 1)/2$
 (d) more than $n(n - 1)/2$
14. The maximum degree of any vertex in a simple graph with n vertices is
 (a) $n - 1$ (b) $n + 1$
 (c) $2n - 1$ (d) N
15. One can convert a binary tree into its mirror image by traversing it in
 (a) inorder (b) preorder
 (c) postorder (d) any order
16. The number of leaf nodes in a complete binary tree of depth d is
 (a) $2d$ (b) $2d - 1 + 1$
 (c) $2d + 1 + 1$ (d) $2d + 1$
17. What data structure would you mostly likely see in a non-recursive implementation of a recursive algorithm?
 (a) Stack (b) Linked list
 (c) Queue (d) Trees
18. A graph with n vertices will definitely have a parallel edge or self-loop if the total number of edges are
 (a) greater than $n - 1$
 (b) less than $n(n - 1)$
 (c) greater than $n(n - 1)/2$
 (d) less than $n^2/2$
19. The pre-order and post order traversal of a Binary Tree generates the same output. The tree can have maximum
 (a) Three nodes (b) Two nodes
 (c) One node (d) Any number of nodes
20. The postfix form of $A * B + C / D$ is
 (a) $*AB / CD +$ (b) $AB * CD / +$
 (c) $A * BC + / D$ (d) $ABCD + / *$

Section B: Short Answer Questions

[55 Marks]

Question 01

- i. What do asymptotic notation means and what are the basic asymptotic efficiency classes? (2)
- ii. What is worst-case, average-case, best- case efficiency of an algorithm? (3)

- iii. How does an array differ from an ordinary variable? (1)
- iv. A stack is to be implemented using an array. The associated declarations are:

```
int stack [100];
int top = 0;
```

 Give the statement to perform push operation. (2)
- v. Assume that a queue is available for pushing and popping elements. Given an input sequence a, b, c, (c be the first element), give the output sequence of elements if the rightmost element given above is the first to be popped from the queue. (1)
- vi. What is the maximum possible number of nodes in a binary tree at level 6? (1)

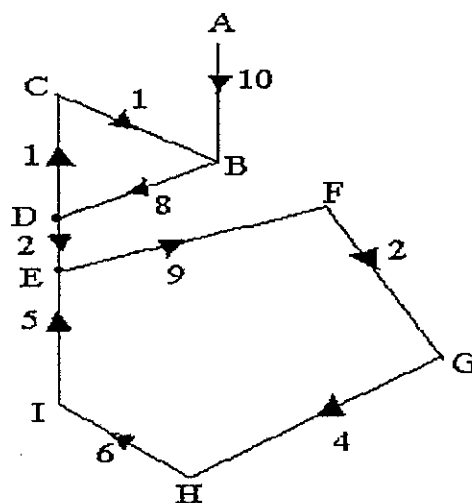
Question 2

- i. What is an algorithm? What are the characteristics of a good algorithm? (2)
- ii. How do you find the complexity of an algorithm? What is the relation between the time and space complexities of an algorithm? Justify your answer with an example. (3)
- iii. Define the term array. How are two-dimensional arrays represented in memory? Explain how address of an element is calculated in a two dimensional array. (3)
- iv. Draw the expression tree of the following infix expression. Convert it in to Prefix and Postfix expressions. (7)

$$((a + b) + c * (d + e) + f) * (g + h)$$

Question 3

- i. Give the adjacency matrix for the following graph: (3)



- ii. Write an algorithm to insert a node in the beginning of the linked list. (3)

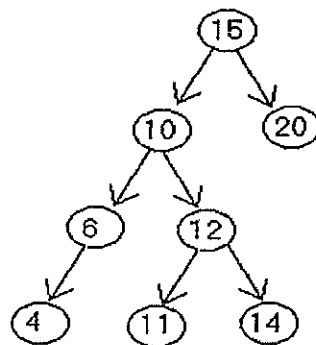
- iii. Convert the following infix expressions into its equivalent postfix and prefix expressions: (4)
- $(A + B \wedge D) / (E - F) + G$
 - $A * (B + D) / E - F * (G + H / K)$

Question 4

- Which are the two standard ways of traversing a graph? Explain them with an example of each. (3)
- Consider the following specification of a graph G
 $V(G) = \{1, 2, 3, 4\}$
 $E(G) = \{(1, 2), (1, 3), (3, 3), (3, 4), (4, 1)\}$
 - Draw an undirected graph. (1.5)
 - Draw its adjacency matrix. (1.5)
- What do you mean by hashing? Explain some popular hash functions. (3)
- How do collisions happen during hashing? (1)

Question 5

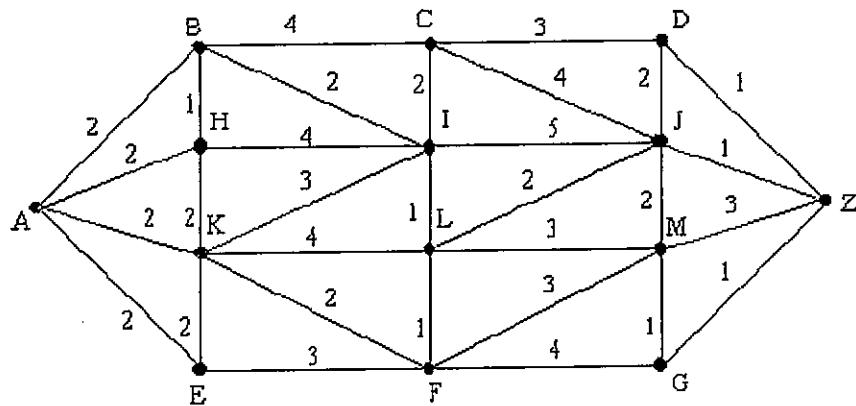
- Define data type and abstract data type. Comment upon the significance of both. (2)
- Write an algorithm to delete a particular node from binary search tree. Trace your algorithm to delete a node (10) from the given tree. (3)



- Define a linked-list? How are these stored in the memory? Suppose the linked list in the memory consisting of numerical values. Write a procedure to calculate the sum of the values stored in the list. (3)
- Big oh notation characterizes the asymptotic behaviour of a function by providing an upper bound on the rate at which the function grows as n gets large. How can we test to see whether a given asymptotic bound is the best one available? Explain with an example. (2)

Section C: Long Answer Questions (25 marks)

- i. Let P be a pointer to a singly linked list. Show how this list may be used as a stack. That is, write algorithms to push and pop elements. Specify the value of P when the stack is empty. (5)
- ii. What is a Binary Search Tree (BST)? Make a BST for the following sequence of numbers.
45, 36, 76, 23, 89, 115, 98, 39, 41, 56, 69, 48
Traverse the tree in Preorder, Inorder and postorder. (5)
- iii. Find the shortest path from A to Z using Dijkstra's Algorithm. (5)
Hint: You don't have to visit all the vertices from A → Z.



- iv. Consider the following eight numbers 50, 33, 44, 22, 77, 35, 60 and 40. Display the construction of the binary by inserting the above numbers in the given order. (5)
- v. Write a program in terms of an array that would be size of 6 and it should store its values in its index as its square values for example:
Numbers[0] = 0
Numbers[1] = 1
Numbers[2] = 4
Numbers[3] = 9
Numbers[4] = 16
Numbers[5] = 25

Write appropriate methods to show this together with a print method. Also include these methods with a call in the main program. (5)
Hint: you may use the code skeleton given below:

using System;

```
class SingleDimArrayApp
{
    protected int[] numbers;

    SingleDimArrayApp()
    {
        numbers = new int[6];
        for ( condition )
        {
```

```

        // statements;
    }
}

protected void PrintArray()
{
    for (conditions)
    {
        // statements
    }
}

public static void Main()
{
    // write method call here
    Console.ReadLine();
}
}

```

THE END!!

