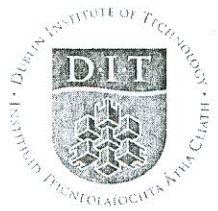

DUBLIN INSTITUTE OF TECHNOLOGY

DT211C BSc. (Honours) Degree in Computer Science (Infrastructure)

Year 1

SUMMER EXAMINATIONS 2015/2016

INTRODUCTION TO ALGORITHMS [CMPU1014]

DR. BASEL MAGABLEH
DR. DEIRDRE LILLIS

FRIDAY 13TH MAY 4.00 – 6.00PM

2 HOURS

INSTRUCTIONS TO CANDIDATES

ANSWER **THREE** QUESTIONS OUT OF **FOUR**.

ALL QUESTIONS CARRY EQUAL MARKS.
EACH STUDENT WILL BE AWARDED A BONUS OF 1 MARK

Question 1

a) Define what is meant by the terms

(i) *Algorithms,*

(ii) *Complete Binary Tree,*

(iii) *Perfect Binary Tree,*

(iv) *Doubly-linked list,*

(v) *Shell sort*

(15 marks)

b) Describe how one can implement each of the following operations on an □array so that the time it takes does not depend on the array's size `n`

(i) Delete the `ith` element of an array ($1 \leq i \leq n$)

(ii) Delete the `ith` element of a sorted array (the remaining array has to stay sorted, of course). □

(10 marks)

c) List clearly all the steps of the Selection Sort algorithm for sorting an array A[] of size n. Show how this algorithm sorts the following array in increasing order, by showing the sequence of values after each pass through the array:

E X A M P L E

(8 marks)

Question 2

a) Consider the following algorithm.

```
ALGORITHM Secret(A[0..n - 1])
//Input: An array A[0..n - 1] of n real numbers
minval ← A[0]; maxval ← A[0] for i ← 1 to n - 1 do
if A[i] < minval minval ← A[i]
if A[i] > maxval maxval ← A[i]
return maxval - minval.
```

Answer the following questions:

1. What does this algorithm compute?
2. What is its basic operation?
3. How many times is the basic operation executed?
4. What is time complexity for this algorithm in Big-O notation?

(12 marks)

b) Consider the following algorithm.

```
ALGORITHM Enigma(A[0..n - 1, 0..n - 1])
//Input: A matrix A[0..n - 1, 0..n - 1] of real
numbers
for i ← 0 to n - 2 do
for j ← i + 1 to n - 1 do
if A[i, j] <> A[j, i]
return false
return true
```

Answer the following questions:

1. What does this algorithm compute?
2. What is its basic operation?
3. How many times is the basic operation executed?
4. What is time complexity for this algorithm in Big-O notation?

(12 marks)

c) Write down the `pseudocode` for the two algorithms for *enqueing* `void Enqueue(int x)` and *dequeing* *void* `Dequeue()` on a queue that is implemented using a linked list

(7 marks)

Question 3

a) With the help of a stack, evaluate the following expressions given in postfix form. Show the stack contents after each operation:

```
a) 5 9 8 * 4 6 * * 7 + *
b) 300 23 + 43 21 - * 84 7 + /
c) 4 8 + 6 5 - * 3 2 - 2 2 + * /
```

(12 marks)

b) Insert the following integers into a binary tree, in the order given:

```
a b c d e f
```

Write down the (i) pre-order and (ii) in-order (iii) post-order traversals of the resulting tree.

(12 marks)

c) Draw a binary tree with 10 nodes labeled 0,1,...,9 in such a way that the inorder and postorder traversals of the tree yield the following lists: 9, 3, 1, 0, 4, 2, 7, 6, 8, 5 (inorder) and 9, 1, 4, 0, 3, 6, 7, 5, 8, 2 (postorder).

(7 marks)

Question 4

a) Write down `pseudocode`, with the help of a diagram, to add a new element to the tree showing in Figure 1. You need to place the new item in the correct order.

(10 marks)

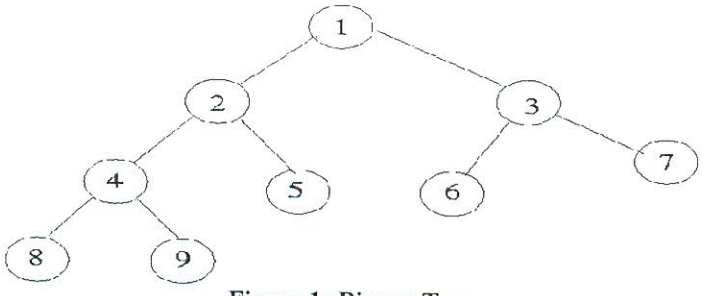

Figure 1: Binary Tree

b) Write down the `pseudocode` of an algorithm for checking whether two given words are anagrams, i.e., whether one word can be obtained by permuting the letters of the other. For example, the words `tea` and `eat` are anagrams.

(12 Marks)

c) Write down the `pseudocode` for the `push(stack, item)` and `pop(stack)` functions for a stack implemented using an array with `size` elements.

(11 Marks)