CAMPBELL UNIVERSITY

NORTH CAROLINA, U. S. A.

ACADEMIC YEAR 2015/2016

AUGUST/SEPTEMBER EXAMINATION

COMPUTER SCIENCE BACS2063(B)
DATA STRUCTURES AND ALGORITHMS

WEDNESDAY, 26 AUGUST 2015 TIME: 2.00 PM – 3.00 PM (1 HOUR)

BACHELOR OF SCIENCE DEGREE

Instructions to Candidates:

Answer **ALL** questions. All questions carry equal marks.

BACS2063(B) DATA STRUCTURES AND ALGORITHMS

Q1. (a) Given the array:

0	1	2	3	4	5	6
Sandra	Betty	Peter	Danny	Helen	Kent	Moses

Apply the following sorting techniques to sort the array in alphabetical order. Show the contents of the array after each pass is performed.

(i) Insertion sort (6 marks)

(ii) Selection sort (6 marks)

(b) Explain **ONE(1)** type of traversal for binary trees. You are required to explain the steps involved in the traversal. (6 marks)

(c) Write a *sequential search algorithm* for an unsorted array. The algorithm should return true if the target value is found in the array and false otherwise. (7 marks)

[Total: 25 marks]

Q2. (a) A sorted list may be implemented using a *linked list* with a *head reference*. Given an ordered list (Figure 1) with the following names:

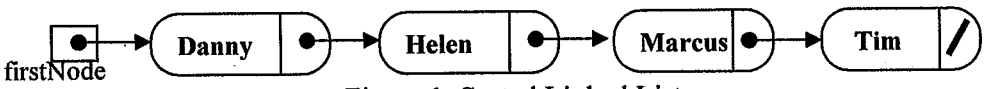

Figure 1: Sorted Linked List

(i) Write a recursive method `displayBackward(Node firstNode)` to displays the contents of a chain of linked nodes in reversed order.

For example, the output for the linked list in Figure 1 should be Tim Marcus Helen Danny. (6 marks)

Given the `Node` class definition:

```
public class Node {
   private T    data;
   private Node next;

   private Node(T dataPortion) {
      data = dataPortion;
      next = null;
      } // end constructor

   private Node(T dataPortion, Node nextNode) {
      data = dataPortion;
      next = nextNode;
      } // end constructor
 } // end Node
```

Q2.(a)(Continued)

(ii) Comment and elaborate on the *memory efficiency* of the method `displayBackward` method. (4 marks)

(iii) Remove the name *Tim* from the sorted list. Explain the time efficiency in Big-Oh notation. (4 marks)

(b) The following keys are hashed into a *hash table of size 7*:
17, 23, 35, 47, 58, 71

Given the primary hash function
h_1(key) = key modulo 7

and that collisions are resolved by *double hashing* with the secondary hash function
h_2(key) = 7 - key modulo 11

Required:
- Show your working for the hash indexes for each key.
- Draw the resulting hash table after all the keys have been inserted.

(5+6 marks)
[Total: 25 marks]