Comp 2140 Data Structures and Algorithms

Cuneyt Akcora   Midterm   March 9 2020
E2-105, EITC   Version 2   9:30 – 10:20 a.m.

PRINT Student Name: _____

Signature: _____

Student ID: _____

Directions:

1- No aids permitted: no books, no paper, no notes, no electronic devices (including no calculators, no translation devices, no computers, no cell phones, no mp3 players, no smart watches, no tablets, and so on).

2- Legibly write each answer in the space provided after the question.

3- Page 8 is extra space for the answer to any question (if you need it).

4- Page 9 contains classes needed for the last two questions. You may remove it if you wish.
5- Use the backs of pages for rough work.

The exam is out of 30 marks and has 9 pages.

Question	Mark	Out of
1		6
2		4
3		6
4		4
5		4
6		4

1- **[6 marks]** Suppose we run three sorting methods on copies of an array containing **1, 7, 19, 24, 7, 49** (in that order). Suppose we stop each method before it is complete (as described below). In each case, give the order of the values as they would appear in the array after we stopped each sort.

     a. Quick sort (that uses the median-of-three technique to find its pivot) stopped IMMEDIATELY BEFORE the first recursive call starts.

     b. Radix sort stopped IMMEDIATELY AFTER the first iteration of the outermost loop.

     c. Recursive merge sort stopped IMMEDIATELY AFTER the first recursive call is finished.

2- [**4 marks**]

The following table lists the values returned and the stack that results from a series of stack operations beginning with an empty stack (bottom [> top). For each of the eight incomplete rows, fill in the stack operation that must have been performed to produce the value returned (if any) and resulting stack.

Resulting stack	Value returned	Stack operation
[>	—	Stack created
[1 >	—	
[>	1	
[>	true	
[5 >	—	
[5 7 >	—	
[5 >	7	
[5 >	5	
[5 3 >	—	

3- [**6 marks**] Write a recursive method addEvens that recursively adds the even numbers in an int array

completely filled with positive integers, and returns their sum.

Examples: if the input array contains 2, 12, 3, 4, 3, 5, then the output is $2 + 12 + 4 = 18$ and if the input is 1,

3, 5, then the output is 0 because the input contains no even numbers. (More space on the next page if you

need it.)

4- [**4 marks**] Which abstract data type(s) that should we prefer when data insertion order must be preserved?

5- [4 marks] Give the computational complexity (worst case running time) of the following code snippets.

    Express your answer with the O-notation and be as precise as possible.

```void function1 (int n) { int q = 1; for (int w=1; w < n; w++) for (int j=1; j < w; j++) q = q*j; }```	
```void function2 (int n) { int q = 1; for (int w=1; w < n; w++) q = q*j; }```	

6- [4 marks] Consider the linked list given in the following code snippet.

Write the output of this class here:_____

```java
import java.util.Random;

public class ShortLinkedList {
    public class Node {
        public int item;
        public  Node next;

        public Node(int newItem, Node newNext) {
            item = newItem;
            next = newNext;
        }
    }
    private  Node top;
    private static Random generator = new Random(System.nanoTime());

    public ShortLinkedList() {
        top = null;
    }

    private void insertValue(int i){
        Node newNode = new Node(i,top);
    }

    private boolean isEmpty(){
        boolean res = false;
        if(top==null) res=true;
        return res;
    }

    public static void main(String[] args) {

        ShortLinkedList list1 = new ShortLinkedList();
        boolean b = (list1.top==null);
        list1 =change(list1);
        boolean c = (list1.top==null);
        System.out.println(b+","+c);
    }

    public static ShortLinkedList change(ShortLinkedList l1){
        l1 = new ShortLinkedList();
        l1.insertValue(3);
        l1.insertValue(4);
        return l1;
    }
}
```