Started on	Monday, 7 October 2024, 8:33 PM
State	Finished
Completed on	Monday, 7 October 2024, 8:56 PM
Time taken	23 mins 3 secs
Marks	17.00/20.00
Grade	**85.00** out of 100.00

Question **1**

Complete

Mark 1.00 out of 1.00

What is the output of the following program?

```
1: public class Laptop {
2: public void start() {
3: try {
4: System.out.print("Starting up ");
5: throw new Exception();
6: } catch (Exception e) {
7: System.out.print("Problem ");
8: System.exit(0);
9: } finally {
10: System.out.print("Shutting down ");
11: }
12: }
13: public static void main(String[] args) {
14: new Laptop().start();
15: } }
```

Select one or more:

- [ ] **a. Starting up Problem Shutting down**
- [ ] **b. The code does not compile.**
- [ ] **c. Starting up Shutting down**
- [ ] **d. An uncaught exception is thrown.**
- [x] **e. Starting up Problem**
- [ ] **f. Starting up**

Question 2

Complete

Mark 0.00 out of 1.00

What is the output of the following code snippet?

```
3: boolean x = true, z = true;
4: int y = 20;
5: x = (y != 10) ^ (z=false);
6: System.out.println(x+", "+y+", "+z);
```

Select one or more:

- [ ] **a. false, 20, false**
- [x] **b. The code will not compile because of line 5.**
- [ ] **c. true, 20, false**
- [ ] **d. false, 20, true**
- [ ] **e. false, 20, true**
- [ ] **f. true, 10, true**

Question 3

Complete

Mark 1.00 out of 1.00

What will be the output of the following Java program for recursion?

```
public class RecursionExample {
    public static int calculate(int n) {
        if (n == 0) {
            return 0;
        } else {
            return n + calculate(n - 2);
        }
    }
       public static void main(String[] args) {
        System.out.println(calculate(9));
    }
}
```

Select one or more:

- [ ] **a. 25**
- [x] **b. stack overflow error**
- [ ] **c. 30**
- [ ] **d. 20**

Question **4**

Complete

Mark 1.00 out of 1.00

What is the output of the following code?

```
1: package rope;
2: public class Rope {
3: public static int LENGTH = 5;
4: static {
5: LENGTH = 10;
6: }
7: public static void swing() {
8: System.out.print("swing ");
9: }
10: }
```

```
1: import rope.*;
2: import static rope.Rope.*;
3: public class Chimp {
4: public static void main(String[] args) {
5: Rope.swing();
6: new Rope().swing();
7: System.out.println(LENGTH);
8: }
9: }
```

Select one or more:

- [ ] **a. swing swing 5**
- [ ] **b. Compiler error on line 2 of Chimp.**
- [ ] **c. Compiler error on line 5 of Chimp.**
- [ ] **d. Compiler error on line 7 of Chimp.**
- [ ] **e. Compiler error on line 5 of Chimp.**
- [x] **f. swing swing 10**

Question 5

Complete

Mark 1.00 out of 1.00

Consider the following pseudo code for deleting a node with a given value from a singly linked list:

```
function deleteNode(head, value):
    if head is NULL:
        return NULL
    if head.data == value:
        return head.next
    current = head
    while current.next != NULL and current.next.data != value:
        current = current.next
    if current.next == NULL:
        return head
    current.next = current.next.next
    return head
```

What is the time complexity of this algorithm in the worst case?

Select one or more:

- [x] **a. O(n)**
- [ ] **b. O(log n)**
- [ ] **c. O(1)**
- [ ] **d. O(n^2)**

Question 6

Complete

Mark 1.00 out of 1.00

Choose the correct statement about the following code:

```
1: public interface CanFly {
2: void fly();
3: }
4: interface HasWings {
5: public abstract Object getWindSpan();
6: }
7: abstract class Falcon implements CanFly, HasWings {
8: }
```

Select one or more:

- [ ] **a. The code will not compile because of line 4.**
- [ ] **b. The code will not compile because of line 5.**
- [ ] **c. The code will not compile because of line 2.**
- [x] **d. It compiles without issue.**
- [ ] **e. The code will not compile because of lines 2 and 5.**
- [ ] **f. The code will not compile because the class Falcon doesn't implement the interface methods.**

Question 7

Complete

Mark 0.00 out of 1.00

What is the output of the following pseudo-code for inserting an element into a binary search tree (BST)?

```
class TreeNode {
    int data;
    TreeNode left, right;
}
// Function to insert an element into the BST
function insert(TreeNode root, int value) {
    if (root == null) {
        return new TreeNode(value);
    }
    if (value < root.data) {
        root.left = insert(root.left, value);
    } else if (value > root.data) {
        root.right = insert(root.right, value);
    }
    return root;
}
```

If the following values are inserted into the tree in order: 10, 5, 20, 3, 7, 15, what will be the structure of the binary search tree?

Select one or more:

- [ ] **a.**
```
    15
    / \
   10  20
   / \
  7   5
```
- [ ] **b.**
```
    10
    / \
   7   20
  / \  /
 5  3 15
```
- [x] **c.**
```
    10
    / \
   5   15
  / \   \
 3   7   20
```

d.
```
    10
   / \
  5   20
 / \  /
3  7 15
```

Question 8

Complete

Mark 1.00 out of 1.00

Two pipes A and B can fill a tank in 20 minutes and 30 minutes respectively. Another pipe C can empty the full tank in 25 minutes. If all three pipes are opened together, how much time will it take to fill the tank?

Select one or more:

- a. 22 minutes
- b. 26 minutes
- c. 25 minutes
- d. 23 minutes

Question 9

Complete

Mark 1.00 out of 1.00

A shopkeeper marks the price of an article 40% above the cost price and gives a discount of 20%. What is the profit percentage?

Select one or more:

- a. 10%
- b. 15%
- c. 8%
- d. 12%

Question 10

Complete

Mark 1.00 out of 1.00

A train 180 meters long is moving at a speed of 72 km/h. How long will it take to cross a platform 120 meters long?

Select one or more:

- a. 10 seconds
- b. 18 seconds
- c. 15 seconds
- d. 12 seconds

Question 11

Complete

Mark 1.00 out of 1.00

Evaluate the following prefix expression using a stack:

+ * 5 4 - 10 / 8 2

What is the result of this expression?

Select one or more:

- [ ] **a. 32**
- [ ] **b. 42**
- [x] **c. 26**
- [ ] **d. 28**

Question 12

Complete

Mark 1.00 out of 1.00

Given the following class in the file /my/directory/named/A/Bird.java:

INSERT CODE HERE

public class Bird { }

Which of the following replaces INSERT CODE HERE if we compile from /my/directory?

(Choose all that apply)

Select one or more:

- [ ] **a. Does not compile.**
- [ ] **b. package A;**
- [ ] **c. package a;**
- [ ] **d. package my.directory.named.a;**
- [x] **e. package named.A;**
- [ ] **f. package named.a;**
- [ ] **g. package my.directory.named.A;**

Question 13

Complete

Mark 1.00 out of 1.00

In a circular queue implementation using an array, suppose the queue has a size of 5 and front = 2, rear = 4. After enqueuing one more element, what would be the value of rear?

Select one or more:

- [x] **a. 0**
- [ ] **b. 1**
- [ ] **c. 5**
- [ ] **d. 3**

Question 14

Complete

Mark 1.00 out of 1.00

Two trains 250 meters and 350 meters long are running on parallel tracks in the same direction at speeds of 60 km/h and 45 km/h, respectively. How long will it take for the faster train to completely pass the slower train?

Select one or more:

- [x] **a. 144 seconds**
- [ ] **b. 150 seconds**
- [ ] **c. 172 seconds**
- [ ] **d. 160 seconds**

Question 15

Complete

Mark 1.00 out of 1.00

Which of the following cases in an AVL tree requires a left rotation to balance the tree?

Select one or more:

- [x] **a. Insertion in the right subtree of the right child of a node.**
- [ ] **b. Insertion in the right subtree of the left child of a node.**
- [ ] **c. Insertion in the left subtree of the right child of a node.**
- [ ] **d. Insertion in the left subtree of the left child of a node.**

Question 16

Complete

Mark 1.00 out of 1.00

Two trains are moving in opposite directions with speeds of 60 km/h and 90 km/h, respectively. If they take 12 seconds to cross each other, and the length of the first train is 150 meters, what is the length of the second train?

Select one or more:

- [ ] **a. 380 meters**
- [ ] **b. 270 meters**
- [ ] **c. 300 meters**
- [x] **d. 350 meters**

Question 17

Complete

Mark 1.00 out of 1.00

Which are the results of the following code? (Choose all that apply)

```
String numbers = "012345678";
System.out.println(numbers.substring(1, 3));
System.out.println(numbers.substring(7, 7));
System.out.println(numbers.substring(7));
```

Select one or more:

- [x] **a. 78**
- [x] **b. 12**
- [ ] **c. An exception is thrown.**
- [ ] **d. The code does not compile.**
- [ ] **e. 7**
- [x] **f. A blank line.**
- [ ] **g. 128**

Question 18

Complete

Mark 1.00 out of 1.00

Consider the following pseudo code for merging two sorted linked lists:

```
function merge(list1, list2):
    if list1 == NULL:
        return list2
    if list2 == NULL:
        return list1
    if list1.data < list2.data:
        list1.next = merge(list1.next, list2)
        return list1
    else:
        list2.next = merge(list1, list2.next)
        return list2
```

What is the time complexity of merging two sorted linked lists of sizes m and n?

Select one or more:

- [ ] **a. O(m log n)**
- [x] **b. O(m + n)**
- [ ] **c. O(n log m)**
- [ ] **d. O(m * n)**

Question 19

Complete

Mark 1.00 out of 1.00

If we insert the following sequence of values into an empty BST: 50, 30, 70, 20, 40, 60, 80, what will be the inorder traversal of the resulting BST?

Select one or more:

- [x] **a. 20 30 40 50 60 70 80**
- [ ] **b. 50 20 30 40 60 70 80**
- [ ] **c. 50 30 20 40 70 60 80**
- [ ] **d. 80 70 60 50 40 30 20**

Question **20**

Complete

Mark 0.00 out of 1.00

The following pseudo-code is used to find the middle element of a singly linked list. What is the missing condition in the while loop?

```
class Node {
    int data;
    Node next;
}
// Function to find the middle of the linked list
function findMiddle(Node head) {
    Node slow = head;
    Node fast = head;
    while (/* Missing Condition */) {
        slow = slow.next;
        fast = fast.next.next;
    }
    return slow.data;
}
```

What should be the condition in the while loop?

Select one or more:

- [x] **a. fast != null && fast.next != null && slow != null**
- [ ] **b. fast != null**
- [ ] **c. fast != null && fast.next != null**
- [ ] **d. slow != null && slow.next != null**