

**Solution to Test of Computer Science 203x Level**

Score: \_\_\_\_\_ / 100 Time: 100 Minutes

**PART I: Multiple Choice (3 points each)**

Note: The correct answer can be any combination of A, B, C, D.

Example,

Sample Question: which one is int type? ( ABD )

A. -1                      B. 2                      C. 2.5                      D. 9

Question 1:

What's the value of variable a after executing the follow statement? (B)

String a = 5 \* 3 + "2" + (9 / 2 + 13 % 3);

A. "22"                      B. "1525"                      C. "152(9/2+13%3)"                      D. "125"

Question 2:

What's the result of the following expression? (C)

 $7.0/2 + 2 * 4$ 

A. 10                      B. 10.5                      C. 11.5                      D. 11

Question 3:

Which following statement is correct for static method? (CD)

- A. In order to call the static method, the instance of the class that contains the static method must be created.
- B. Within a static method, it can call other non-static method.
- C. Within a non-static method, it can call other static method.
- D. The static method can be called without instantiating an object of the class that contains the static method.

Question 4:

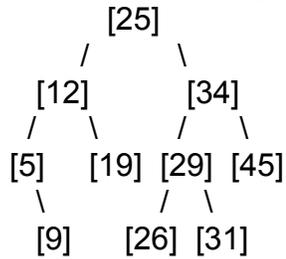
```
public class House extends Building {
    private int height;
    private int width;
    public House (int height, int width) {
        /* Constructor */
    }
}
```

Which code in the constructor of House is correct? (AC)

- A. this.height = height;  
this.width = width;
- B. this.height = height;  
this.width = width;  
super();
- C. super();  
this.height = height;  
this.width = width;
- D. this.height = height;  
super();  
this.height = height;

Question 5 ~ 8 share the same tree.

Assume there is a binary search tree like:



Question 5:

What's the pre-order traversal visiting order?

(C)

- A. 9, 5, 19, 12, 26, 31, 29, 45, 34, 25
- B. 25, 12, 34, 5, 19, 29, 45, 9, 26, 31
- C. 25, 12, 5, 9, 19, 34, 29, 26, 31, 45
- D. 5, 9, 12, 19, 25, 26, 29, 31, 34, 45
- E. None of those

Question 6:

What's the in-order traversal visiting order?

(D)

- A. 9, 5, 19, 12, 26, 31, 29, 45, 34, 25
- B. 25, 12, 34, 5, 19, 29, 45, 9, 26, 31
- C. 25, 12, 5, 9, 19, 34, 29, 26, 31, 45
- D. 5, 9, 12, 19, 25, 26, 29, 31, 34, 45
- E. None of those

Question 7:

What's the post-order traversal visiting order?

(A)

- A. 9, 5, 19, 12, 26, 31, 29, 45, 34, 25
- B. 25, 12, 34, 5, 19, 29, 45, 9, 26, 31
- C. 25, 12, 5, 9, 19, 34, 29, 26, 31, 45
- D. 5, 9, 12, 19, 25, 26, 29, 31, 34, 45
- E. None of those

Question 8:

What's the BFS traversal visiting order?

(B)

- A. 9, 5, 19, 12, 26, 31, 29, 45, 34, 25
- B. 25, 12, 34, 5, 19, 29, 45, 9, 26, 31
- C. 25, 12, 5, 9, 19, 34, 29, 26, 31, 45
- D. 5, 9, 12, 19, 25, 26, 29, 31, 34, 45
- E. None of those

Question 9:

What's the data structure we used in breath first search?

(C)

- A. Stack
- B. ArrayList
- C. Queue
- D. Heap
- E. LinkedList

Question 10:

What's the data structure we used in depth first search?

(A)

- A. Stack
- B. ArrayList
- C. Queue
- D. Heap
- E. LinkedList

**PART II: Output (5 points each)**

Question 1:

Write the output of following code:

```
System.out.print("n/n/n\n\n");  
System.out.print("\\\\");
```

n/n/n

\\

Question 2:

Write the output of following code:

```
for (int i = 0; i < 3; i++) {  
    for (int j = 0; j < i - 3; j++) {  
        System.out.print("A");  
    }  
    for (int k = 0; k < i; k++) {  
        System.out.print("E");  
    }  
    System.out.println();  
}
```

E  
EE

Question 3:

We have a pre-defined static method:

```
public static void secret(int num, int limit) {  
    if (num < limit) {  
        System.out.print("a");  
        secret(num + 1, limit);  
    } else {  
        System.out.print("b");  
    }  
}
```

Write the output of calling secret(2, 4);

aab

Name: \_\_\_\_\_

Question 4:

We have pre-defined static methods:

```
public static void method1(int num) {  
    method2(num - 1);  
    System.out.println("method1");  
}
```

```
public static void method2(int num) {  
    if (num > 0) {  
        method2(num - 1);  
        System.out.println("method2");  
    }  
}
```

```
public static void method3() {  
    System.out.println("method3");  
    method2(1);  
    method1(3);  
}
```

Write the output of calling method3()

```
method3  
method2  
method2  
method2  
method1
```

Name: \_\_\_\_\_

Question 5:

Assume that the following classes have been defined:

```
public class Mouse extends Rat {
    public void method2() {
        System.out.print("a 2 ");
        method1();
    }
}

public class Rat extends Animal {
    public String toString() {
        return "b";
    }

    public void method2() {
        System.out.print("b 2 ");
        super.method2();
    }
}
```

```
public class Animal {
    public String toString() {
        return "c";
    }

    public void method1() {
        System.out.print("c 1 ");
    }

    public void method2() {
        System.out.print("c 2 ");
    }
}

public class Hamster extends Rat {
    public void method1() {
        System.out.print("d 1 ");
        method2();
    }
}
```

Given the classes above, what output is produced by the following code?

```
Animal[] animals = {new Mouse(), new Rat(), new Animal(), new Hamster()};
for (int i = 0; i < animals.length; i++) {
    System.out.println(animals[i]);
    animals[i].method1();
    System.out.println();
    animals[i].method2();
    System.out.println();
}
```

```
b
c 1
a 2 c 1
b
c 1
b 2 c 2
c
c 1
c 2
b
d 1 b 2 c 2
b 2 c 2
```

**PART III: Short Answer (5 points each)**

Question 1:

Convert an unsigned decimal number 23 to a binary number.

**10111**

Question 2:

Use binary search to find 99 in a sorted array

{2, 5, 21, 44, 53, 61, 82, 89, 91, 99, 120}

1st value to visit in the array: **61**2nd value to visit in the array: **91**3rd value to visit in the array: **99**4th value to visit in the array: **N/A**

Question 3:

A linked list's head is given. The linked list is shown as below:

head -&gt; [5] -&gt; [8] -&gt; [9] -&gt; [12] -&gt; null

Write code to remove the fourth node [12] from the linked list?

**head.next.next.next = null;****or head.next.next.next = head.next.next.next.next;**

Question 4:

A linked list's head is given. The linked list is shown as below:

head -&gt; [4] -&gt; [8] -&gt; [20] -&gt; [58] -&gt; null

```
class Node {
    private int value;
    public Node next;
    public Node(int val) {
        value = val;
    }
}
```

Write code to add [7] before the second node [8] and after the first node [4]?

**Node tmp = new Node(7);****tmp.next = head.next;****head.next = tmp;**

Question 5:

A linked list's head is given. The linked list is shown as below:

head -&gt; [71] -&gt; [12] -&gt; [13] -&gt; [24] -&gt; [64] -&gt; ... -&gt; [89] -&gt; [21] -&gt; [21] -&gt; [22] -&gt; null

Write code to remove all nodes whose values are odd numbers.

**Node current = head;**

```
while (current.next != null) {
    if (current.next.value % 2 == 1) {
        current.next = current.next.next;
    } else {
        current = current.next;
    }
}
if (head.value % 2 == 1) {
    head = head.next;
}
```

**PART IV: Programming (10 points each)**

Question 1:

A given array that contains integer numbers that are disordered such as {8, 7, 11, 10, 21, 55, 21}. Write a static method to accept the given array and return true if there are duplicate values otherwise return false

For example, given {8, 7, 11, 10, 21, 55, 21} => return true

given {8, 7, 10, 10, -21, 55, 21} => return true

given {8, 7, 11, 10, 20, 55, 21} => return false

Solution 1:

```
public void static boolean isDuplicate(int[] numbers) {
    if (numbers != null && numbers.length > 1) {
        Set<Integer> visitedNumbers = new HashSet<>();
        for (int i = 0; i < numbers.length; i++) {
            if (visitedNumbers.contains(numbers[i])) {
                return true;
            } else {
                visitedNumbers.add(numbers[i]);
            }
        }
    }
    return false;
}
```

Solution 2:

```
public void static boolean isDuplicate(int[] numbers) {
    if (numbers != null && numbers.length > 1) {
        Set<Integer> uniqueNumbers = new HashSet<>();
        for (int i = 0; i < numbers.length; i++) {
            uniqueNumbers.add(numbers[i]);
        }
        return uniqueNumbers.size() == numbers.length;
    }
    return false;
}
```

## Question 2:

A given array that contains integer numbers that are disordered such as {0, 8, 2, 11, 10, 21, 55}. Write a static method to accept the given array and print all the pairs such that their sum is equal to 10. (Duplicates may be in the array and each number in the array at most can be printed once.)

```
public static void printAllPairs(int[] numbers) {
    if (numbers != null && numbers.length > 0) {
        Map<Integer, Integer> numberToOccurrence = new HashMap<>();
        for (int i = 0; i < numbers.length; i++) {
            if (numberToOccurrence.containsKey(10 - numbers[i])
                && numberToOccurrence.get(10 - numbers[i]) > 0) {
                System.out.println(numbers[i] + " " + (10 - numbers[i]));
                numberToOccurrence.put(10 - numbers[i],
                    numberToOccurrence.get(10 - numbers[i]) - 1);
            } else {
                int updatedOccurrence = 1;
                if (numberToOccurrence.containsKey(numbers[i])) {
                    updatedOccurrence +=
                        numberToOccurrence.get(numbers[i]);
                }
                numberToOccurrence.put(numbers[i],
                    updatedOccurrence);
            }
        }
    }
}
```