Answer 1)

```
public class SquareRoot {
    public static int mySqrt(int x) {
        if (x == 0 || x == 1) {
            return x;
        }

        long left = 0;
        long right = x;

        while (left <= right) {
            long mid = left + (right - left) / 2;

            if (mid * mid <= x) {
                left = mid + 1;
            } else {
                right = mid - 1;
            }
        }

        return (int) right;
    }

    public static void main(String[] args) {
        int x = 4;
        int result = mySqrt(x);

        System.out.println("Square root of " + x + " is " + result);
    }
}
```

Answer 2)

```
class ListNode {
    int val;
    ListNode next;

    ListNode(int val) {
        this.val = val;
    }
}
```

```
public class AddTwoNumbers {
    public ListNode addTwoNumbers(ListNode l1, ListNode l2) {
        ListNode dummy = new ListNode(0);
        ListNode current = dummy;
        int carry = 0;

        while (l1 != null || l2 != null) {
            int sum = carry;
            if (l1 != null) {
                sum += l1.val;
                l1 = l1.next;
            }
            if (l2 != null) {
                sum += l2.val;
                l2 = l2.next;
            }

            current.next = new ListNode(sum % 10);
            carry = sum / 10;
            current = current.next;
        }

        if (carry > 0) {
            current.next = new ListNode(carry);
        }

        return dummy.next;
    }

    public static void main(String[] args) {
        // Example 1: l1 = [2,4,3], l2 = [5,6,4]
        ListNode l1 = new ListNode(2);
        l1.next = new ListNode(4);
        l1.next.next = new ListNode(3);

        ListNode l2 = new ListNode(5);
        l2.next = new ListNode(6);
        l2.next.next = new ListNode(4);

        AddTwoNumbers solution = new AddTwoNumbers();
        ListNode result = solution.addTwoNumbers(l1, l2);
        printLinkedList(result);
```

```
    }

    private static void printLinkedList(ListNode head) {
        ListNode current = head;
        while (current != null) {
            System.out.print(current.val + " ");
            current = current.next;
        }
        System.out.println();
    }
}
```