Module IN2002—Data Structures and Algorithms
Exercise Sheet 4

1. Suppose we have `Node p` and `Node q` referring to the same list of nodes containing `[5, 8, 3]`. Draw the list, and the results of the following statements, applied in order:

```
p.next = null;
p = null;
```

2. Suppose we have `Node p` referring to a list of nodes containing `[3, 5, 4, 1]`. Draw the list, and the results of the following statements, applied in order:

```
p.next = p.next.next;
p.next = new Node(7, p.next);
p.next.next = new Node(8);
p.next.next = p;
```

3. Work out (using pictures) what the following procedure does:

```
public void modify(SLList list) {
     if (list.head != null && list.head.next != null) {
          Node tmp = list.head.next;
          list.head.next = tmp.next;
          tmp.next = list.head;
          list.head = tmp;
     }
}
```

4.
a) Write a method that checks whether two singly linked lists have the same elements (in any order). You should not modify either list. What time (in big-O notation) does this method take?

b) What would be the time complexity if you were checking whether the lists have the same elements in the same order?