# Lab 8. List

## Part 1: Single Liked List

The questions in this part all relate to the singly linked list data structure. Complete the methods in the **SinglyLinkedList** class below based on the Node class and the List interface in the notes. The **print** method is included to help you when you are testing the implementation.

### Question 1

Complete the implementation of the insertFirst and insertLast methods. Remember that these methods should return the Position (Node) where the data was inserted.

### Question 2

Complete the implementation of the insertAfrter and insertBefore methods. Remember that these methods should return the Position (Node) where the data was inserted.

### Question 3

Complete the implementation of the remove method. This should remove the node from the list and return the data that was stored inside it.
Remember to think and account for any special cases (such as being the last element in the list). You can assume that the Position passed as a parameter is a Node from the list.

### Question 4

Write a separate class to do some testing. You should test each of the methods you are required to implement.

- Your tests should test all possible usages of every method
- All results should be commented and printed out

**Submission**:

One Zip file including One Java Files named: *SinglyLinkedList.java*

# Part 2: Array Based List

The questions in this part all relate to the array based list data structure. Complete the methods in the **ArrayList** class and the List and Position Interfaces in notes. The **print** method is included to help you when you are testing the implementation.

## Question 1

Modify the code for the **last** method so that it checks if the list is empty before returning the value

## Question 2

Complete the implementation of the **insertFirst** and **insertLast** methods. Remember that these methods should return the Position (ArrPos) where the data was inserted.

## Question 3

Complete the implementation of the **insertAfrter** and **insertBefore** methods. Remember that these methods should return the Position (ArrPos) where the data was inserted.

## Question 4

Complete the implementation of the **remove** method. This should remove the node from the list and return the data that was stored inside it.
Remember to think and account for any special cases (such as being the last element in the list). You can assume that the Position passed as a parameter is an element from the list.

## Question 5

Write a separate class to do some testing. You should test each of the methods you are required to implement.

- Your tests should test all possible usages of every method
- All results should be commented and printed out

**Submission**:

One Zip file including One Java Files named: *ArrayList.java*