

CS 314 FINAL REVIEW — LINKED LIST SUMS

Linked Lists & Recursion

Implement an instance method for the singly-linked `IntLinkedList` class which changes a nodes value to be the sum of its own value plus the value of all subsequent nodes in the list.

Complete the following method.

```
// Changes each node's int value to be the sum of itself plus the sum of
// all the values which come after it in the list.
// pre: size > 0
// post: size is unchanged
public void addValuesToRight() {
```

Here are some example calls to `addValuesToRight()`:

```
[1, 1, 1, 1, 1].addValuesToRight() => [5, 4, 3, 2, 1]
[1, 2, 3, 4, 5].addValuesToRight() => [15, 14, 12, 9, 5]
[0, 0, 0, 0, 2].addValuesToRight() => [2, 2, 2, 2, 2]
[5, 1, 2, 0].addValuesToRight() => [8, 3, 2, 0]
[3, 1, 4].addValuesToRight() => [8, 5, 4]
[7].addValuesToRight() => [7]
```

You may use the following `IntLinkedList` implementation:

```
public class IntLinkedList {
    IntNode first;
    int size;

    //Nested node class
    private static class IntNode{
        IntNode next;
        int value;
    }
}
```

Your solution should be as efficient as possible in terms of time.

You may create helper methods.

Do not create any new data structures or use any other Java classes or methods.

```
// Changes each node's int value to be the sum of itself plus the sum of
// all the values which come after it in the list.
// pre: size > 0
// post: size is unchanged
public void addValuesToRight() {
```