

## CS 0445 Spring 2024 Assignment 2

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

**Note 1:** All methods must be implemented in an efficient way, utilizing the underlying linked list. Methods that have repeated unnecessary accesses to Nodes (ex: using getNodeAt() or getEntry() to iterate through the list) and other inefficient implementations will receive minimal credit. In these cases credit will be deducted BOTH from individual method line items AND from the "Methods implemented in efficient way" item.

**Note 2:** Correctness of methods includes handling of special cases

### LinkedListPlus:

Constructor (from Recitation 4): \_\_\_\_\_ (10)

leftShift, rightShift(): \_\_\_\_\_ (10)

leftRotate(), rightRotate(): \_\_\_\_\_ (10)

No new Nodes created in above methods: \_\_\_\_\_ (10)

LLPTest.java works with no changes: \_\_\_\_\_ (10)

### ReallyLongInt:

toString(): \_\_\_\_\_ (5)

constructor (for long arg): \_\_\_\_\_ (10)

add(): \_\_\_\_\_ (25)

subtract(): \_\_\_\_\_ (25)

multiply(): \_\_\_\_\_ (25)

compareTo(): \_\_\_\_\_ (15)

equals(): \_\_\_\_\_ (5)

RLITest.java works with no changes: \_\_\_\_\_ (10)

Methods implemented in efficient way: \_\_\_\_\_ (20)

Documentation: \_\_\_\_\_ (5)

<b>Assignment Information Sheet/Submission:</b>	_____ (5)
<b>Subtotal:</b>	_____ (200)
<b>Normalized Subtotal (Subtotal / 2.0):</b>	_____ (100)
<b>On-time Bonus:</b>	_____ (5)
<b>Late Penalty:</b>	_____ (-5)
<b>Extra Credit:</b>	_____ (10)
<b>Total</b>	_____ <b>(100)</b>