Jeff Bahns

Assignment 3

Long Integer Addition – Algorithm

**Program Flow**

My program will convert inputted numbers into two separate doubly linked lists. The inputted numbers are taken in the form of strings, and will be converted into separate single digit integers during this process.

Starting with each tail of the separate lists, I find the sum. If the sum exceeds 9, I carry over to the next link (but actually the previous link/node).

This takes place inside a while loop. The conditions of the while loop are that each link in both of the lists may not be null.

If it gets to a point where one list still has links and the other is null (the numbers are unequal lengths), then the loop will keep running and adding the remaining numbers to the resultant linked list.

Once again, the resultant sum will be sent into a third linked list. This third linked list is absorbing the integers as nodes simultaneously while the addition is taking place.

**Methods/Classes**

There will be three separate classes two within a master class (one file).

Anything not notated “Class” is a method

* Class Assignment3
* Class Node
  + Node(int d)
    - Standard constructor that takes data input and stores it into class variable data
* Class DoublyLinkedList
  + DoublyLinkedList()
    - Standard constructor that initializes a null head/tail
  + isEmpty()
    - Checks if list is empty
  + insertHead(int data)
    - Inserts a node at beginning of list
  + insertTail(int data)
    - Inserts a node at end of list
  + printList()
    - Prints the entirety of the linked list
* main(String[]args)
  + Drives the program by grabbing input, creating objects, and calling methods.
* convertToList(int number)
  + Converts a long integer (in the form of a string) into a double inked list that contains integers
* listSum(DoubleLinkedList a, DoublyLinkedList b)
  + Takes two doubly linked lists that have integer contents and finds and returns their sum.