**Union and Intersection of Two Linked Lists**

Your task for this problem is to fill out the union and intersection functions. The union of two sets A and B is the set of elements which are in A, in B, or in both A and B. The intersection of two sets A and B, denoted by A ∩ B, is the set of all objects that are members of both the sets A and B.

You will take in two linked lists and return a linked list that is composed of either the union or intersection, respectively. Once you have completed the problem you will create your own test cases and perform your own run time analysis on the code.

We have provided a code template below, you are not required to use it:

**class** **Node**:

**def** **\_\_init\_\_**(self, value):

self.value = value

self.next = **None**

**def** **\_\_repr\_\_**(self):

**return** str(self.value)

**class** **LinkedList**:

**def** **\_\_init\_\_**(self):

self.head = **None**

**def** **\_\_str\_\_**(self):

cur\_head = self.head

out\_string = ""

**while** cur\_head:

out\_string += str(cur\_head.value) + " -> "

cur\_head = cur\_head.next

**return** out\_string

**def** **append**(self, value):

**if** self.head **is** **None**:

self.head = Node(value)

**return**

node = self.head

**while** node.next:

node = node.next

node.next = Node(value)

**def** **size**(self):

size = 0

node = self.head

**while** node:

size += 1

node = node.next

**return** size

**def** **union**(llist\_1, llist\_2):

*# Your Solution Here*

**pass**

**def** **intersection**(llist\_1, llist\_2):

*# Your Solution Here*

**pass**

*# Test case 1*

linked\_list\_1 = LinkedList()

linked\_list\_2 = LinkedList()

element\_1 = [3,2,4,35,6,65,6,4,3,21]

element\_2 = [6,32,4,9,6,1,11,21,1]

**for** i **in** element\_1:

linked\_list\_1.append(i)

**for** i **in** element\_2:

linked\_list\_2.append(i)

**print** (union(linked\_list\_1,linked\_list\_2))

**print** (intersection(linked\_list\_1,linked\_list\_2))

*# Test case 2*

linked\_list\_3 = LinkedList()

linked\_list\_4 = LinkedList()

element\_1 = [3,2,4,35,6,65,6,4,3,23]

element\_2 = [1,7,8,9,11,21,1]

**for** i **in** element\_1:

linked\_list\_3.append(i)

**for** i **in** element\_2:

linked\_list\_4.append(i)

**print** (union(linked\_list\_3,linked\_list\_4))

**print** (intersection(linked\_list\_3,linked\_list\_4))