**Homework #1 progress**

**Hewr Tarkhany**

**add first method**

1), make a new node NN and store the Object item into the new node NN's data field.

2) NN's next reference points to the node referenced by head.next.

3) Then assign NN to the next reference in the dummy head node.

4) increase size of this list by one.

**remove first method**

1. check is size is smaller than zero if yes throw and exception

2. create a nn and assign it to this.head.next

3.then this.head.next assign to nn.next to pu the new node in the right spot

4.decrement the size

5. retun the data of the node that was removed

**contain method**

1. check if the head is null in another word if it is an empty list

2.creat a for look in that creat a new node name cur assign it to this.head.next

3. walk it through till cur.next == null by incrementing cur=cur.next

4.if statement if the object we want to find is null and cur.data is nulll (OR) cur.data .equal(objct) return true

5 .else return false

**Boolean remove(Object o) method**

1. create NEW NODE ListNode cur =this.head.next assign it to the head.next

2. create another node call prev to assign it to the one before cur which is this.head

3. we check with while loop till we get into the end of the list by going prev behind cur and cur=cur.next

4.first we check with if statement if the object that past in is null or the cur.data is null then we skip cur by saying prev.next = cur.next and decrementing the size and returning the true if this is the case

5. second if neither data is null then we see if cur.data equals to the object if yes we remove that node and decrement the size by one. If this case is not true then in the end we return false for the method

**Remove all copies method**

1. First we create a boolean variable and we assign it to the remove method
2. Second we create a while loop saying if remove method is true by passing in the object that’s past in then keep doing the remove method till all the copies are gone
3. Then we return the boolean variable that we created

**Interleave method**

1. We create a linkedlist variable
2. We create 2 nodes and assign them to the head. Next of the 2 nodes that were past in in the method
3. While loop checks if those nodes we created **are not** null if yes then add data into them by calling the add method
4. Then assign each one of them to its next node to go trough all the data inside of them
5. Then we check by if statement if which one is bigger or smaller if one is smaller then add the rest of the bigger on in to the linkedlist variable if not keep going
6. In the end we return the linkedlist variable we created with the data from both nodes we created interleaving

**Add in the index method**

1. We check if the index past in is smaller than zero or it is bigger than the size of the linked list which it will be wrong the we create an exception for that
2. Then we check if the index is 0 which it means if it has one node or list is empty
3. Then we add the object past in by calling the addfirst method to add the data into the first node
4. Then after all checked out if we have a normal list then we create cur and prev nodes to go through the list by calling the for loop statement to check and find the index that is past in to add the node we create into that specific index
5. After we go through the loop then we find the index we create a node called NN then we do prev.next to NN and NN.next to cur.next and then we increment the size by one

**Get object in and index past in method**

1. We check if the index past in is smaller than zero or bigger or equal than the size of the linkedlist if yes then we have an exception to catch that .
2. We create a node cur assign it to the head.next and also integer count to count the index
3. While loop to go through the loop till the end by doing cur= cur.next till cur.next is null
4. If the count equals to the index then return the data of the cur and increment the count integer by one
5. In the end return the cur.data after we found it.

**Remove object by index method**

1. We find if the index is smaller than zero or it is bigger than size if yes then we have an exception to catch that
2. Then we go through the list by for loop to find the index by doing prev = cur and cur= cur.next
3. If the index is zero then we create an object and put the data in the head into it and we also put prev to cur by saying prev = cur
4. And decrement the size and returning the object we created
5. If size is normal then we create and object which we assign it to cur.data
6. Then we asl do prev.next = cur.next by skiping cur to be deleted because at that point cur equals to the node we want to remove
7. Then we decrement the size and return the object that we want to delet

**Add and object in the end of the list method**

1. Create a node named cur and also a NN we assign cur to the head of the list
2. Then we the while loop we say until cur.next is not null then go through the list
3. When the cur.next is null it means we are in the end of the list
4. Then we cur.next = NN
5. Then we increment the size of the list because we are adding a node to the list and of this all worked out we return true because this a boolean method