1. Union()

1. check both of the linked list and add every node value to set {O(n)+ O(n)}

• I use set in order to prevent repetition

2. take the set and make new linked list {O(n)}

Efficiency - 3O(n)-> O(n)

2. Intersection()

1. check the first linked list and add the node value as the key of dictionary {O(n)}

• the value is 0

2. check the second linked list {O(n)}

• if the key is founded in the dictionary , the value is 1

3. take all the key with the value equal to 1 and create linked list O(n)}

Efficiency - 3O(n) -> O(n)