I hereby pledge on my honor that I will strictly adhere to academic integrity codes and the work done on this examination is solely my own and I will not recieve/give any help from/to anybody or source during the examination.

Berke Belgin
17104406

Q1)

a)

```
public boolean removeLast( ) {
    Node<E> node = root;
    if(root == null) return false;
    else if (root.next == null) {
        root.nex = null;
        return true;
    }

    while(node.next.next != null ) {
        node = node.next;
    }
    node.next = null;
    return true;
}
```

1 ←
1 ←
n ←

O(n) (worst-case)

Ω(1) (best-case)

$T(n)_{worst} = 2n+5$

$T(n)_{best} = 3$

b)

```
public boolean removeLast() {
    if(myList.size()==0) return false → 1
    else {
        myList.remove(myList.size()-1); → n (since it is a Linked List)
        return true;
    }
}
```

O(n)

Ω(1)

$T(n)_{worst} = n+2$

$T(n)_{best} = 2$

BB

Berke Belgin
171044065

Q2)
```
public E push(E e) {
        arrList.add(e,0)        → n (since shift operation)
        return e;
}
```

$O(n) = \Omega(n) = \Theta(n)$

```
public E pop() {
    if(arrList.size()==0) return null;   → 1
    E e= arrList.get(arrList.size()-1);
    arrList.remove(0);   → n (since shift operation)
} return e;
```

$O(n)$

$\Omega(1)$