# Cse 222 Homework 02

## Okan Akdoğan

## 121044017

### 1.Problem Definition

In this homework, we need to merge common methods in a interface that named GITList and create two classes for ArrayList & LinkedList which implements GITList interface. In these ArrayList & LinkedList classes we will use KWArrayList and KWLinkedList classes.

### 2.Problem Analysis and Design

Firsly we need create GITList interface wit given methods. After that we can create GITArrayList and GITLinkedList. GITArrayList and GITLinkedList will implements GITList and will use KW classes by composition.

Public interface GITList<E>{ methods }

Public GITArrayList<E> implements GITList<E>{

Private KWArrayList aList;

Methods

Private class itr implements Iterator (for iterator)

Private class listItr extends itr (for list iterator)

}

Public GITLinkedList<E>implements GITList<E>{

Private KWLinkedList lList;

Methods;

}

And we will create Junit classes for two classes

GITArrayListTest and GITLinkedListTest

### 4.Algorithm Analysis in terms of asymtotic notations

#### GITArrayList Methods:

GITArrayList() : O(1)

Add(int index,E obj): O(n)-O(2n) (1 while loop, if reallocate 2 loop)

addAll(GITList<E> l): O(n^2) –O(2n^2)(1 while loop for l, nested loop in add)

AddFirst(E item): O(n)-O(2n) (1 loop add if reallocate 1 more loop )

AddLast(E item): O(n)-O(2n) (1 loop add if reallocate 1 more loop )

ContainsAll(GITList <E> l): O(n^2) (1 loop for l 1 nested loop for arraylist)

Get(int index): O(1)

getFirst(): O(1)

getLast():O(1)

iterator():O(1)

listIterator():O(1)

listIterator(int index):O(1)

RemoveAll(GITList<E> l): O(n^2)(two loops nested)

retainAll(GITList<E> I): O(n^2)(two loops nested)

NESTED Class Itr

Itr(): O(1)

hasNext: O(1)

next():O(1)

remove(): O(n)(for remove and shift loop)

NESTED Class listItr

listItr():O(1)

hasPrevious():O(1)

nextIndex():O(1)

previousIndex:O(1)

previous():O(1)

set(): O(1)

add():O(n)-O(2n)(using GITArrayList add method)

#### GITLinkedList

#### 

Add(int index,E obj): O(n) (listIterator move loop)

addAll(GITList <E> l): O(n^2) (1 loop for list using add loop as nested)

addFirst(E item): O(n)(uses KW class add, 1 loop for iterator)

addLast(E item): O(n)(uses KW class add, 1 loop for iterator)

containsAll(GITList<E> l): O(n^2)(2 nested loops)

get(int index):O(1)

getFirst():O(1)

getLast():O(n) (loop iterator move to end);

iterator():O(1)

listIterator():O(1)

listIterator(int index):O(n) (move to given index)

RemoveAll(GITList<E> l): O(n^2) (2 nested loops)

retainAll(GITList<E> l): O(n^2)(2 nested loops)