Gebze Technical University

Computer Engineering

CSE 222 - 2019 Spring

HOMEWORK 2 REPORT

NEVRA GÜRSES
161044071

Course Assistant: Ayşe Şerbetçi TURAN

1 INTRODUCTION

1.1 Problem Definition

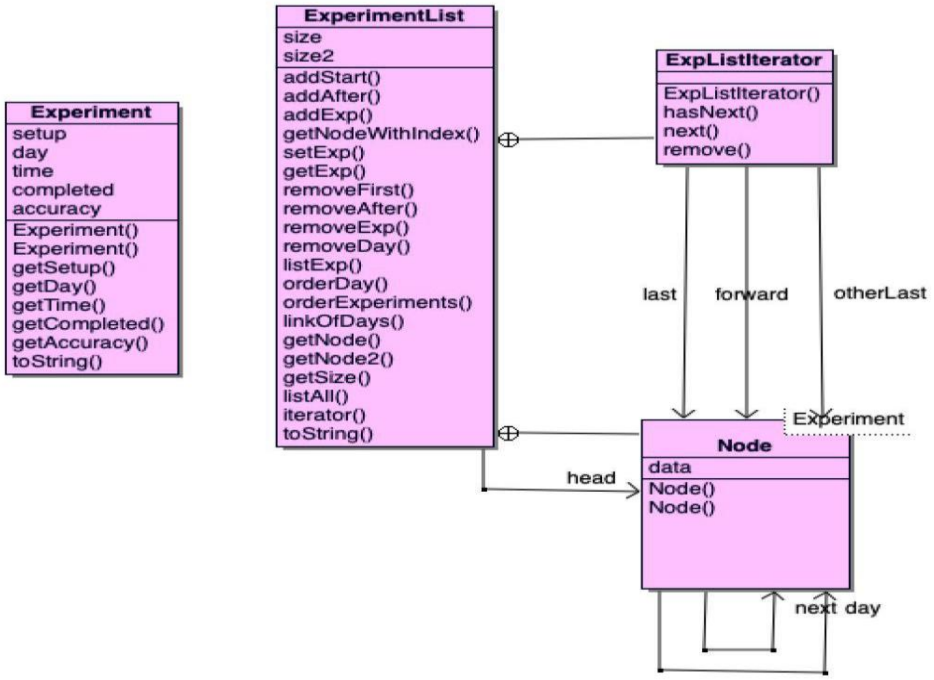
There is experiments that has some features such as start day,experimental setup,start time,state of completeness and output.With these experiments, programmer shoud make a ExprimentList.This ExperimentList implements singly linked list requirements but it must has an additional list structure to accelerate removing and adding operations.Additional list structure should link first experiments of every different day.

1.2 System Requirements

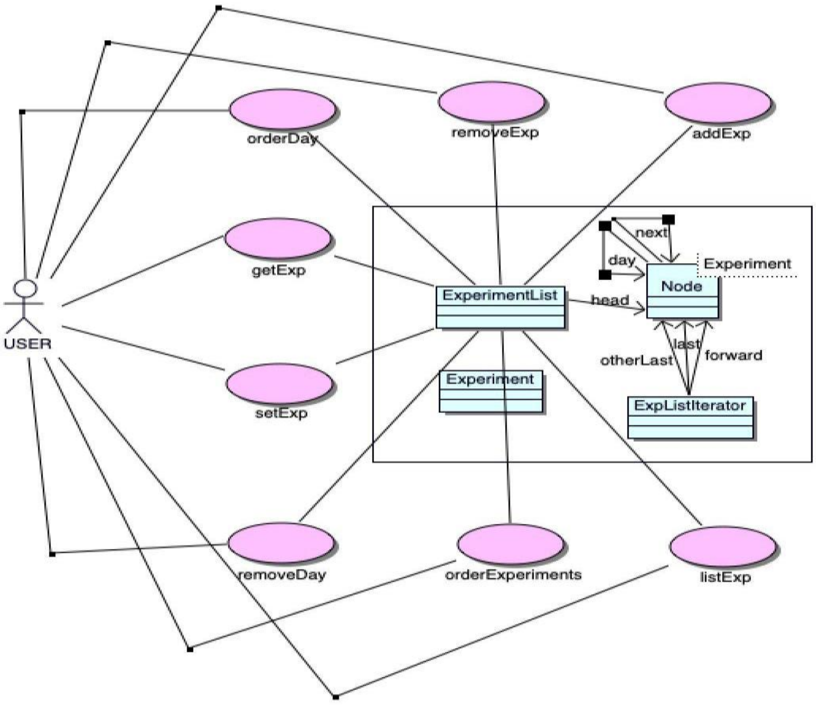
This system require a class for ExperimentList and other class for Experiment features.In order to quick adding and removing operations,system should create a link structure to link first experiments of every different day.Also all experiments should be linked each other by singly linked list.

2 METHOD

2.1 Class Diagrams =>UML DIAGRAM:



2.2 Use Case Diagrams



2.3 Problem Solution Approach

To solve this problem,I create two main class as ExperimentList and Experiment. Experiment class holds features such as day,setup,time etc. of an experiment . ExperimenList class is actual class to create a list. I create also two inner class as Node and ExpListIterator in ExperimetList class.Node has two references that one of them is for next experiment, one of them is for different day.With these next and day references I control day links and experiment links.So I make process such as adding, removing, set etc. in true location according to day, features and position of Experiments.ExpListIerator is requirement in order to be iterable of ExperimentList class.

3 RESULT

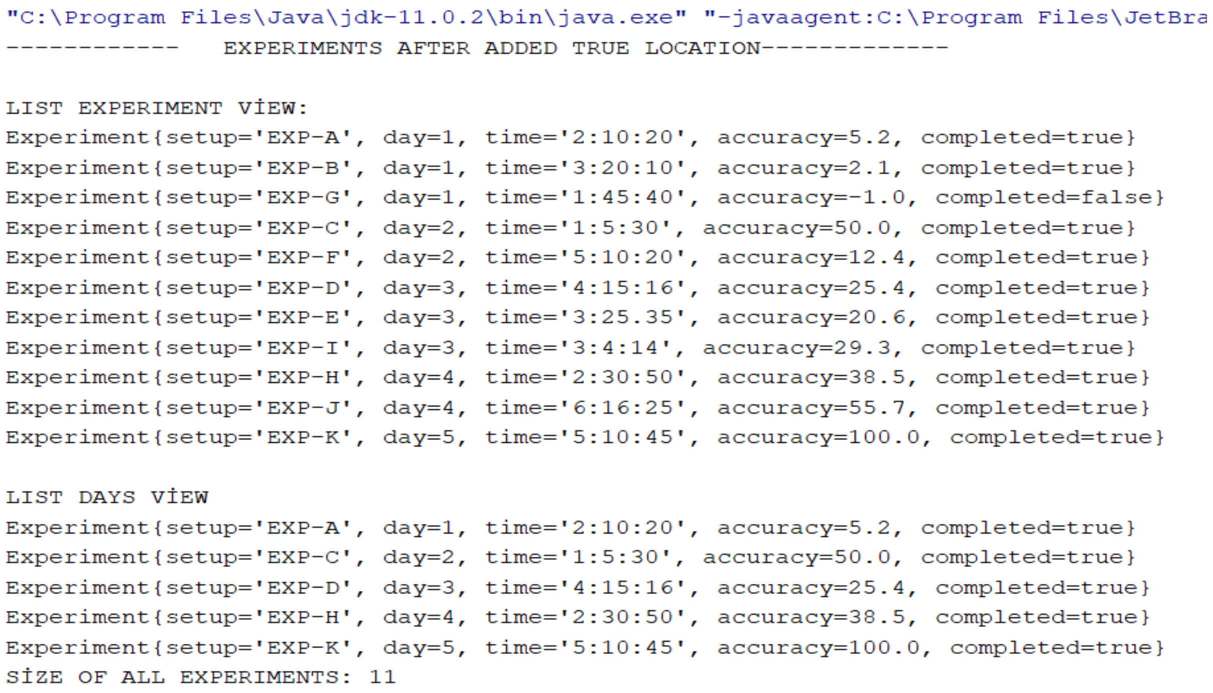
3.1 Test Cases

I added experiments that are different and day of their as: 1-1-2-3-3-2-1-4-3-4-5 in ExperimentList. They are added Experiment list as 1-1-1-2-2-3-3-3-4-4-5.To test removeExp method, I removed day:1,index:0.To test removeDay method,I removed experiments their day is 1. To test set method ,I set experiment day:2,index:1 . To test get method,I get experiment day:3,index:0.To test order method I ordered experiments their day 2. To test listExp method I listed experiments their day:2 .To test iterator() method,I used all experiments in the list.

3.2 Running Results

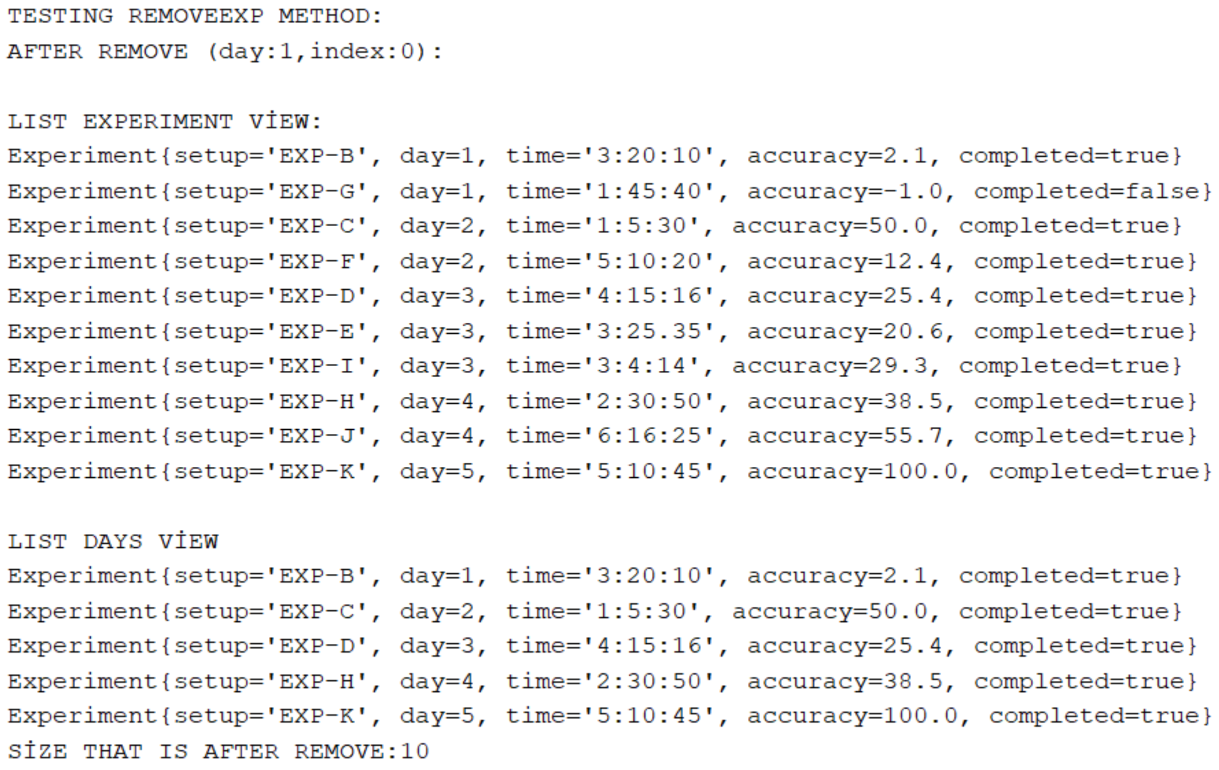
According to my running results, if an experiment will add a list, it is added according to its day.If added experiment is first experiment of any different day, it is linked first experiment of a previous one day.If an experiment will remove from ExperimetList,it is removed according to its day and position.Also,other operations such as set,get,order etc. are made as features of an experiment.

OUTPUT OF ADDEXP METHOD:



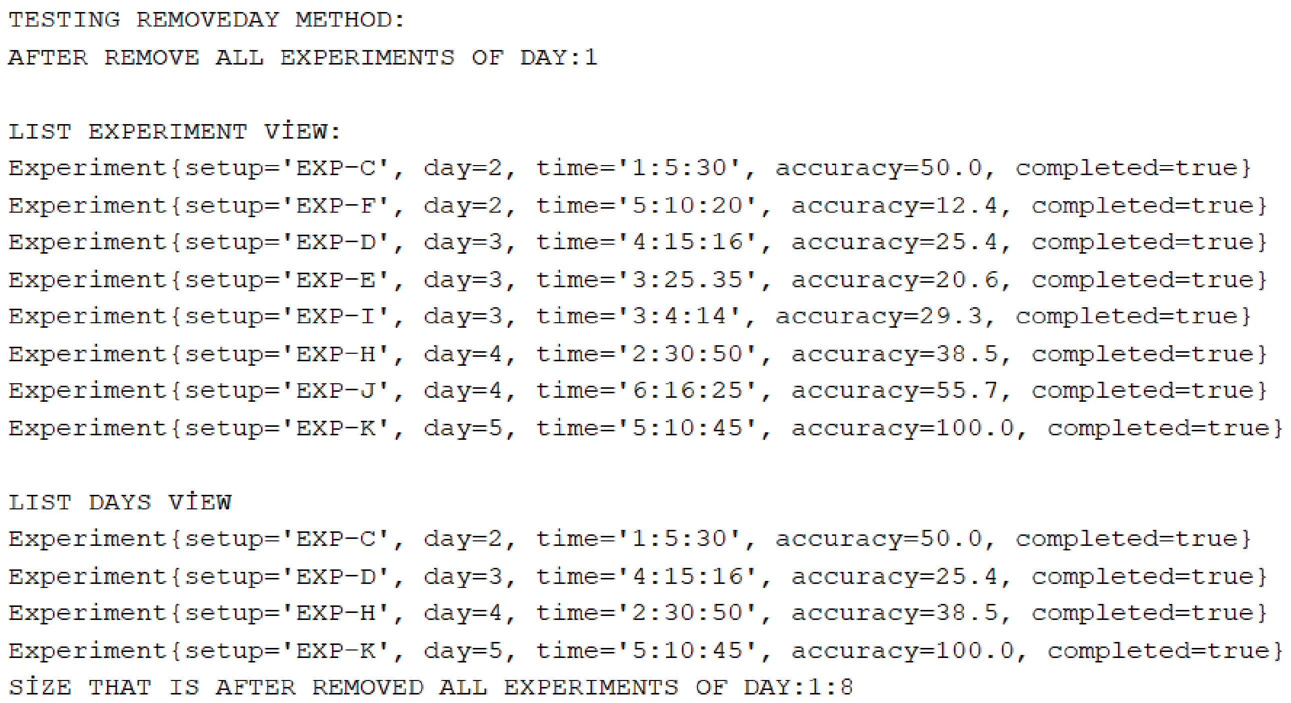
```
"C:\Program Files\Java\jdk-11.0.2\bin\java.exe" "-javaagent:C:\Program Files\JetBra
------------   EXPERIMENTS AFTER ADDED TRUE LOCATION-------------

LIST EXPERIMENT VİEW:
Experiment{setup='EXP-A', day=1, time='2:10:20', accuracy=5.2, completed=true}
Experiment{setup='EXP-B', day=1, time='3:20:10', accuracy=2.1, completed=true}
Experiment{setup='EXP-G', day=1, time='1:45:40', accuracy=-1.0, completed=false}
Experiment{setup='EXP-C', day=2, time='1:5:30', accuracy=50.0, completed=true}
Experiment{setup='EXP-F', day=2, time='5:10:20', accuracy=12.4, completed=true}
Experiment{setup='EXP-D', day=3, time='4:15:16', accuracy=25.4, completed=true}
Experiment{setup='EXP-E', day=3, time='3:25.35', accuracy=20.6, completed=true}
Experiment{setup='EXP-I', day=3, time='3:4:14', accuracy=29.3, completed=true}
Experiment{setup='EXP-H', day=4, time='2:30:50', accuracy=38.5, completed=true}
Experiment{setup='EXP-J', day=4, time='6:16:25', accuracy=55.7, completed=true}
Experiment{setup='EXP-K', day=5, time='5:10:45', accuracy=100.0, completed=true}

LIST DAYS VİEW
Experiment{setup='EXP-A', day=1, time='2:10:20', accuracy=5.2, completed=true}
Experiment{setup='EXP-C', day=2, time='1:5:30', accuracy=50.0, completed=true}
Experiment{setup='EXP-D', day=3, time='4:15:16', accuracy=25.4, completed=true}
Experiment{setup='EXP-H', day=4, time='2:30:50', accuracy=38.5, completed=true}
Experiment{setup='EXP-K', day=5, time='5:10:45', accuracy=100.0, completed=true}
SİZE OF ALL EXPERIMENTS: 11
```

OUTPUT OF REMOVEEXP METHOD:



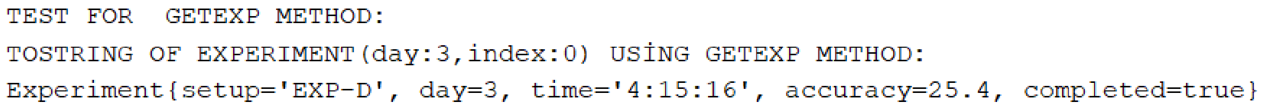
```
TESTING REMOVEEXP METHOD:
AFTER REMOVE (day:1,index:0):

LIST EXPERIMENT VİEW:
Experiment{setup='EXP-B', day=1, time='3:20:10', accuracy=2.1, completed=true}
Experiment{setup='EXP-G', day=1, time='1:45:40', accuracy=-1.0, completed=false}
Experiment{setup='EXP-C', day=2, time='1:5:30', accuracy=50.0, completed=true}
Experiment{setup='EXP-F', day=2, time='5:10:20', accuracy=12.4, completed=true}
Experiment{setup='EXP-D', day=3, time='4:15:16', accuracy=25.4, completed=true}
Experiment{setup='EXP-E', day=3, time='3:25.35', accuracy=20.6, completed=true}
Experiment{setup='EXP-I', day=3, time='3:4:14', accuracy=29.3, completed=true}
Experiment{setup='EXP-H', day=4, time='2:30:50', accuracy=38.5, completed=true}
Experiment{setup='EXP-J', day=4, time='6:16:25', accuracy=55.7, completed=true}
Experiment{setup='EXP-K', day=5, time='5:10:45', accuracy=100.0, completed=true}

LIST DAYS VİEW
Experiment{setup='EXP-B', day=1, time='3:20:10', accuracy=2.1, completed=true}
Experiment{setup='EXP-C', day=2, time='1:5:30', accuracy=50.0, completed=true}
Experiment{setup='EXP-D', day=3, time='4:15:16', accuracy=25.4, completed=true}
Experiment{setup='EXP-H', day=4, time='2:30:50', accuracy=38.5, completed=true}
Experiment{setup='EXP-K', day=5, time='5:10:45', accuracy=100.0, completed=true}
SİZE THAT IS AFTER REMOVE:10
```

OUTPUT OF REMOVEDAY METHOD:



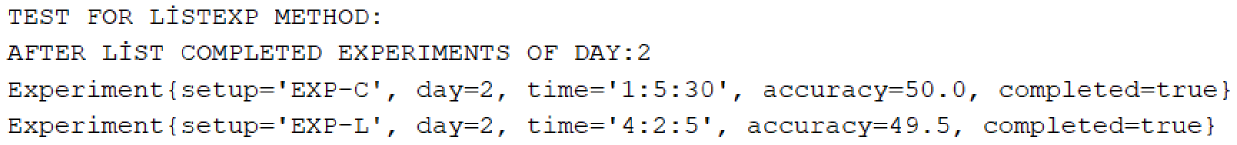
```
TESTING REMOVEDAY METHOD:
AFTER REMOVE ALL EXPERIMENTS OF DAY:1

LIST EXPERIMENT VİEW:
Experiment{setup='EXP-C', day=2, time='1:5:30', accuracy=50.0, completed=true}
Experiment{setup='EXP-F', day=2, time='5:10:20', accuracy=12.4, completed=true}
Experiment{setup='EXP-D', day=3, time='4:15:16', accuracy=25.4, completed=true}
Experiment{setup='EXP-E', day=3, time='3:25.35', accuracy=20.6, completed=true}
Experiment{setup='EXP-I', day=3, time='3:4:14', accuracy=29.3, completed=true}
Experiment{setup='EXP-H', day=4, time='2:30:50', accuracy=38.5, completed=true}
Experiment{setup='EXP-J', day=4, time='6:16:25', accuracy=55.7, completed=true}
Experiment{setup='EXP-K', day=5, time='5:10:45', accuracy=100.0, completed=true}

LIST DAYS VİEW
Experiment{setup='EXP-C', day=2, time='1:5:30', accuracy=50.0, completed=true}
Experiment{setup='EXP-D', day=3, time='4:15:16', accuracy=25.4, completed=true}
Experiment{setup='EXP-H', day=4, time='2:30:50', accuracy=38.5, completed=true}
Experiment{setup='EXP-K', day=5, time='5:10:45', accuracy=100.0, completed=true}
SİZE THAT IS AFTER REMOVED ALL EXPERIMENTS OF DAY:1:8
```

OUTPUT OF GETEXP METHOD:



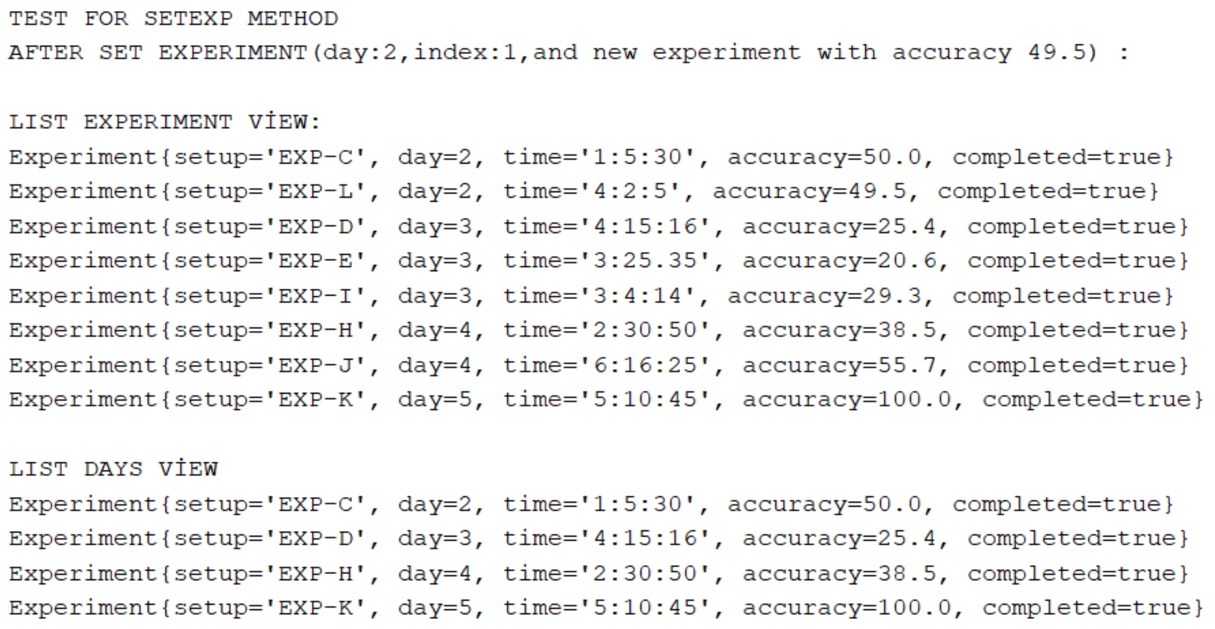
```
TEST FOR  GETEXP METHOD:
TOSTRING OF EXPERIMENT(day:3,index:0) USİNG GETEXP METHOD:
Experiment{setup='EXP-D', day=3, time='4:15:16', accuracy=25.4, completed=true}
```

OUTPUT OF LİSTEXP METHOD:



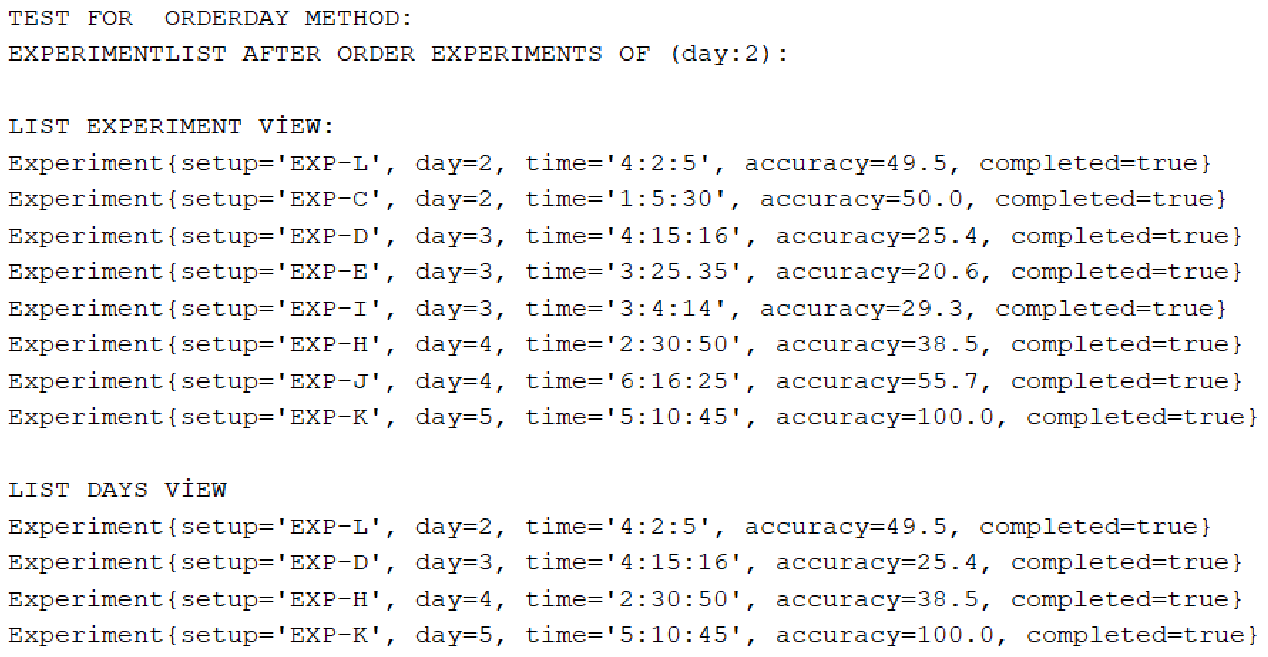
```
TEST FOR LİSTEXP METHOD:
AFTER LİST COMPLETED EXPERIMENTS OF DAY:2
Experiment{setup='EXP-C', day=2, time='1:5:30', accuracy=50.0, completed=true}
Experiment{setup='EXP-L', day=2, time='4:2:5', accuracy=49.5, completed=true}
```

OUTPUT OF SETEXP METHOD:



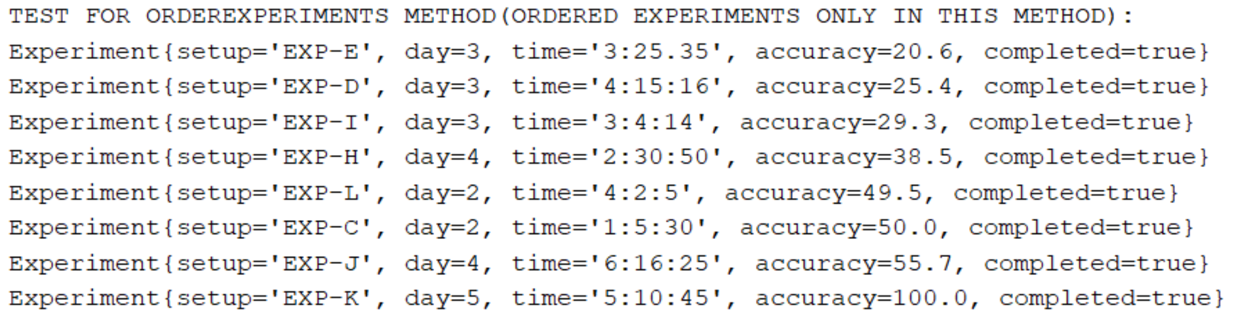
```
TEST FOR SETEXP METHOD
AFTER SET EXPERIMENT(day:2,index:1,and new experiment with accuracy 49.5) :

LIST EXPERIMENT VİEW:
Experiment{setup='EXP-C', day=2, time='1:5:30', accuracy=50.0, completed=true}
Experiment{setup='EXP-L', day=2, time='4:2:5', accuracy=49.5, completed=true}
Experiment{setup='EXP-D', day=3, time='4:15:16', accuracy=25.4, completed=true}
Experiment{setup='EXP-E', day=3, time='3:25.35', accuracy=20.6, completed=true}
Experiment{setup='EXP-I', day=3, time='3:4:14', accuracy=29.3, completed=true}
Experiment{setup='EXP-H', day=4, time='2:30:50', accuracy=38.5, completed=true}
Experiment{setup='EXP-J', day=4, time='6:16:25', accuracy=55.7, completed=true}
Experiment{setup='EXP-K', day=5, time='5:10:45', accuracy=100.0, completed=true}

LIST DAYS VİEW
Experiment{setup='EXP-C', day=2, time='1:5:30', accuracy=50.0, completed=true}
Experiment{setup='EXP-D', day=3, time='4:15:16', accuracy=25.4, completed=true}
Experiment{setup='EXP-H', day=4, time='2:30:50', accuracy=38.5, completed=true}
Experiment{setup='EXP-K', day=5, time='5:10:45', accuracy=100.0, completed=true}
```

OUTPUT OF ORDERDAY METHOD:



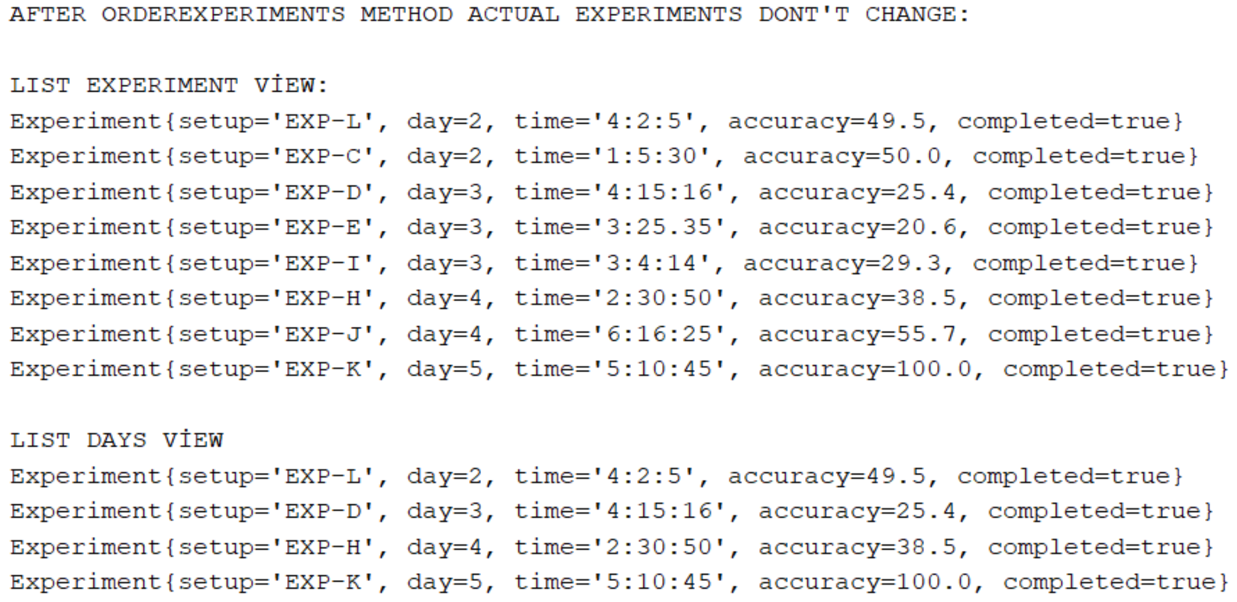
```
TEST FOR  ORDERDAY METHOD:
EXPERIMENTLIST AFTER ORDER EXPERIMENTS OF (day:2):

LIST EXPERIMENT VİEW:
Experiment{setup='EXP-L', day=2, time='4:2:5', accuracy=49.5, completed=true}
Experiment{setup='EXP-C', day=2, time='1:5:30', accuracy=50.0, completed=true}
Experiment{setup='EXP-D', day=3, time='4:15:16', accuracy=25.4, completed=true}
Experiment{setup='EXP-E', day=3, time='3:25.35', accuracy=20.6, completed=true}
Experiment{setup='EXP-I', day=3, time='3:4:14', accuracy=29.3, completed=true}
Experiment{setup='EXP-H', day=4, time='2:30:50', accuracy=38.5, completed=true}
Experiment{setup='EXP-J', day=4, time='6:16:25', accuracy=55.7, completed=true}
Experiment{setup='EXP-K', day=5, time='5:10:45', accuracy=100.0, completed=true}

LIST DAYS VİEW
Experiment{setup='EXP-L', day=2, time='4:2:5', accuracy=49.5, completed=true}
Experiment{setup='EXP-D', day=3, time='4:15:16', accuracy=25.4, completed=true}
Experiment{setup='EXP-H', day=4, time='2:30:50', accuracy=38.5, completed=true}
Experiment{setup='EXP-K', day=5, time='5:10:45', accuracy=100.0, completed=true}
```

OUTPUT OF ORDEREXPERIMENTS METHOD(This method don’ change actual method,I print ordered list only in this function.)



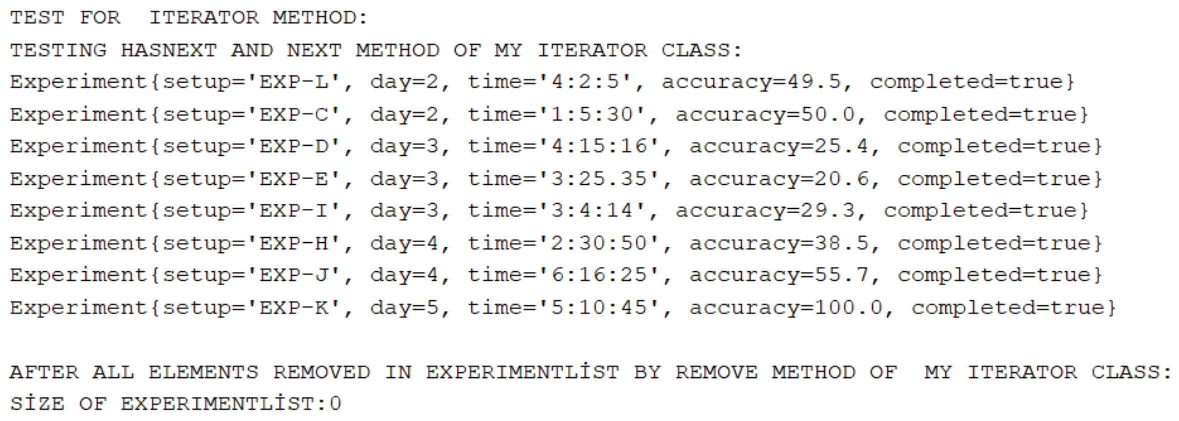
```
TEST FOR ORDEREXPERIMENTS METHOD(ORDERED EXPERIMENTS ONLY IN THIS METHOD):
Experiment{setup='EXP-E', day=3, time='3:25.35', accuracy=20.6, completed=true}
Experiment{setup='EXP-D', day=3, time='4:15:16', accuracy=25.4, completed=true}
Experiment{setup='EXP-I', day=3, time='3:4:14', accuracy=29.3, completed=true}
Experiment{setup='EXP-H', day=4, time='2:30:50', accuracy=38.5, completed=true}
Experiment{setup='EXP-L', day=2, time='4:2:5', accuracy=49.5, completed=true}
Experiment{setup='EXP-C', day=2, time='1:5:30', accuracy=50.0, completed=true}
Experiment{setup='EXP-J', day=4, time='6:16:25', accuracy=55.7, completed=true}
Experiment{setup='EXP-K', day=5, time='5:10:45', accuracy=100.0, completed=true}
```

ORDEREXPERIMENTS METHOD DON’T CHANGE ACTUAL EXPERIMENTS:



```
AFTER ORDEREXPERIMENTS METHOD ACTUAL EXPERIMENTS DONT'T CHANGE:

LIST EXPERIMENT VİEW:
Experiment{setup='EXP-L', day=2, time='4:2:5', accuracy=49.5, completed=true}
Experiment{setup='EXP-C', day=2, time='1:5:30', accuracy=50.0, completed=true}
Experiment{setup='EXP-D', day=3, time='4:15:16', accuracy=25.4, completed=true}
Experiment{setup='EXP-E', day=3, time='3:25.35', accuracy=20.6, completed=true}
Experiment{setup='EXP-I', day=3, time='3:4:14', accuracy=29.3, completed=true}
Experiment{setup='EXP-H', day=4, time='2:30:50', accuracy=38.5, completed=true}
Experiment{setup='EXP-J', day=4, time='6:16:25', accuracy=55.7, completed=true}
Experiment{setup='EXP-K', day=5, time='5:10:45', accuracy=100.0, completed=true}

LIST DAYS VİEW
Experiment{setup='EXP-L', day=2, time='4:2:5', accuracy=49.5, completed=true}
Experiment{setup='EXP-D', day=3, time='4:15:16', accuracy=25.4, completed=true}
Experiment{setup='EXP-H', day=4, time='2:30:50', accuracy=38.5, completed=true}
Experiment{setup='EXP-K', day=5, time='5:10:45', accuracy=100.0, completed=true}
```

OUTPUT OF ITERATOR METHOD:



```
TEST FOR  ITERATOR METHOD:
TESTING HASNEXT AND NEXT METHOD OF MY ITERATOR CLASS:
Experiment{setup='EXP-L', day=2, time='4:2:5', accuracy=49.5, completed=true}
Experiment{setup='EXP-C', day=2, time='1:5:30', accuracy=50.0, completed=true}
Experiment{setup='EXP-D', day=3, time='4:15:16', accuracy=25.4, completed=true}
Experiment{setup='EXP-E', day=3, time='3:25.35', accuracy=20.6, completed=true}
Experiment{setup='EXP-I', day=3, time='3:4:14', accuracy=29.3, completed=true}
Experiment{setup='EXP-H', day=4, time='2:30:50', accuracy=38.5, completed=true}
Experiment{setup='EXP-J', day=4, time='6:16:25', accuracy=55.7, completed=true}
Experiment{setup='EXP-K', day=5, time='5:10:45', accuracy=100.0, completed=true}

AFTER ALL ELEMENTS REMOVED IN EXPERIMENTLİST BY REMOVE METHOD OF  MY ITERATOR CLASS:
SİZE OF EXPERIMENTLİST:0
```

TIME COMPLEXTY OF METHODS:

EXPLANATIONS:

METHODS OF EXPERIMENTLISTCLASS:

1)void addStart(Experiment ex) : Time complexity of it is O(1). It has constant time complexity because there is only assign and increment operations.

2)void addAfter(Node<Experiment> node,Node<Experiment> node2,Experiment ex) : Time complexity of it is O(1).It has constant time complexity because there is two if statement and two of them has O(1) complexity.

3) void addExp(Experiment experiment) : Best case is O(1),because if size==0,addStart method is called and complexity of it is O(1). Worst case is O(n^2) because in the else statement there is inner while loop so if this statement occured,time complexity is O(n^2).Average case is O(n^2) because being possibility of which if statement is1/2. If we add this,1/2O(1) + 1/2O(n^2) =O(n^2).

4) Node<Experiment> getNodeWithIndex(int day,int index):Time complexity of this method is O(n).Because there is a single while loop.

5) void setExp(int day,int index,Experiment e):Time complexity of this method is O(n) because I call getNodeWithIndex method in this method and complexity of it is O(n).

6) Experiment getExp(int day,int index):Time complexity of this method is O(n) because I call getNodeWithIndex method in this method and complexity of it is O(n).

7) Experiment removeFirst() :Time complexity of this method is O(1).It has constant time complexity because there is if - else statement and being posibility of two of them is ½ and two statement has O(1) complexity. 1/2O(1) + ½O(1)= O(1) complexity.

8) Experiment removeAfter(Node<Experiment> node1):Time complexity of this method is O(1). It has constant time complexity because there is if else statement and being posibility of two of them is ½ and two statement has O(1) complexity. 1/2O(1) + ½O(1)= O(1) complexity.

9) void removeExp(int day,int index) : Best case is O(1) because if removing experiment is head removeFirst method is called and it has O(1) complexity.Worst case is O(n) because other statements include one while loop and that while loop has O(n) complexity.Average case is O(n) because 1/3O(1) +1/3O(n) + 1/3O(n) has O(n) complexity.

10) void removeDay(int day): It has O(n) complexity because it has a while loop.

11)void listExp(int day) :Time complexity of this method is O(n^2) because it has inner while loops.

12) void orderDay(int day):It has O(n^2) complexity because it has inner while loops.

13) Node<Experiment> orderExperiments() :It has O(n^2) complexity.Because first while loop has O(n) complexity,other inner while loop has O(n^2) complexity, last while loop has O(n) complexity. Sum of these has O(n^2) complexity.

14) void linkOfDays() :This method has O(n) complexity because it has one while loop.

15) Node<Experiment> getNode(int index): This method has O(n) complexity because it has one while loop.

16) Node<Experiment> getNode2(int index): This method has O(n) complexity because it has one while loop.

17) int getSize():This method has O(1) complexity because it has only return operation.

18) void listAll(): This method has O(n) complexity .it has two while loop.1/2O(n) +1/2O(n) has O(n) complexity.

19) Iterator<Experiment> iterator() :This method O(1) complexity.It has constant time complexity because it only create an iterator and return it.

20) String toString():This method has O(n) complexity.It has a while loop.

METHODS OF EXPERIMENT CLASS:

All methods of this class has O(1) complexity because all methods only have return or assign or print features.

METHODS OF EXPLISTITERATOR CLASS:

All methods of this class has O(1) complexity because their methods only assign or increment operations.These operations is constant time operation.

TIME COMPLEXITY TABLE OF FUNCTIONS:

void addStart(Experiment ex)	O(1)
void addAfter(Node<Experiment> node,Node<Experiment> node2,Experiment ex)	O(1)
void addExp(Experiment experiment)	O(n^2)
Node<Experiment> getNodeWithIndex(int day,int index)	O(n)
void setExp(int day,int index,Experiment e)	O(n)
Experiment getExp(int day,int index)	O(n)
Experiment removeFirst()	O(1)
Experiment removeAfter(Node<Experiment> node1)	O(1)
void removeExp(int day,int index)	O(n)
void removeDay(int day)	O(n)
void listExp(int day)	O(n^2)
void orderDay(int day)	O(n^2)
Node<Experiment> orderExperiments()	O(n^2)
void linkOfDays()	O(n)
Node<Experiment> getNode(int index)	O(n)
Node<Experiment> getNode2(int index)	O(n)
int getSize()	O(1)
void listAll()	O(n)
Iterator<Experiment> iterator()	O(1)
String toString()	O(n)