

**Gebze Technical University
Computer Engineering**

CSE 222 - 2018 Spring

HOMEWORK 1 REPORT

**Rumeysa KARAKAVAK
141044063**

Course Assistant: Ayse Serbetci Turan

1 INTRODUCTION

1.1 Problem Definition

In this work, keeps the track of Experiments for the machine learning system. You can check all the experiment and you can operate them owing to this project.

1.2 System Requirements

For the Experiment:

All experiments have setup for the name, accuracy for the accuracy, time for the beginning of the experiment, day for the beginning of the experiment and a boolean value for the is the experiment completed or not.

setup (String): explains the experimental setup

day(integer): represents the day of start

time(Time): represents the time of start

completed(boolean): indicates whether it is completed or not

accuracy(float): represents the output (not a valid value if the experiment is not completed)

You can change this attributes owing to getter and setter methods.

For the ExperimentList:

We can manage this experiments in the linked list. So we create a new class that implements Iterable class for the manage them. This class has Nodes for each Experiments.

-If you want to add new node <Experiment> on your linked list you must use addExp() method.

-Or you want to remove any node from linked list you must use removeExp() method.

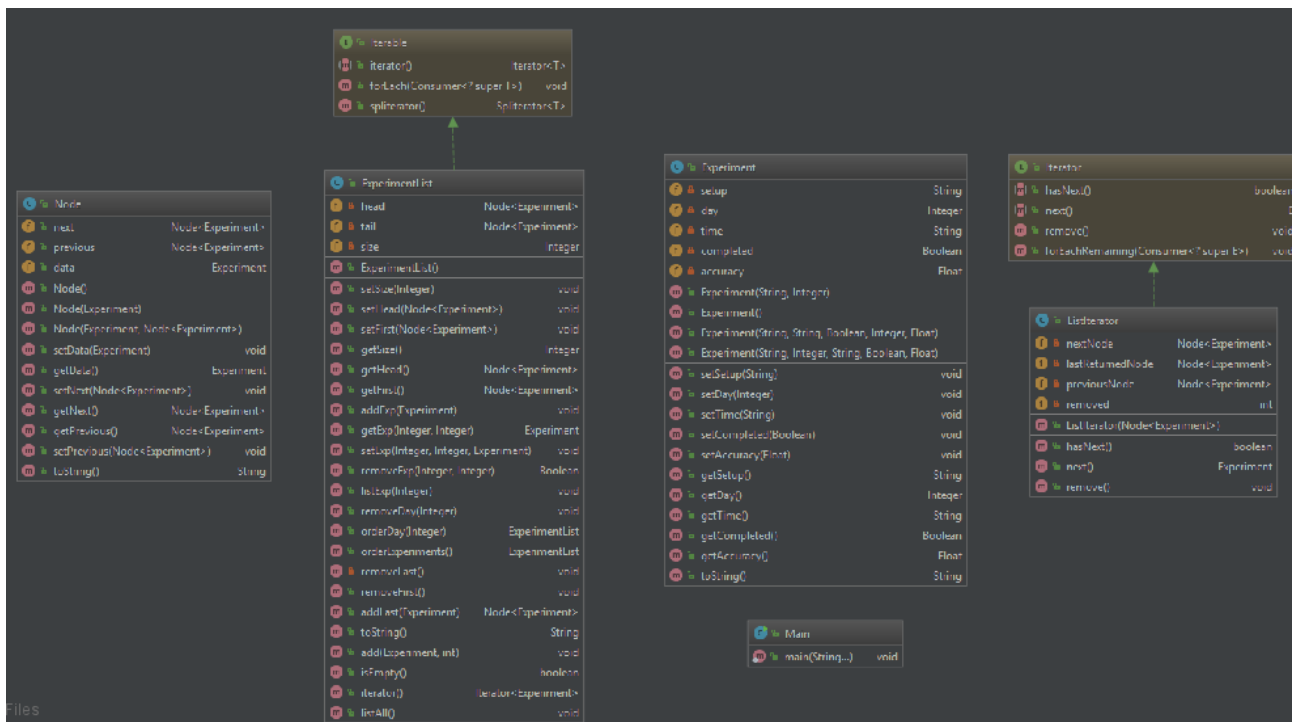
-You can get information about any experiment at given index and day owing to getExp(day, index) method or you can set the information about any experiment at given index and day owing to setExp(day, index, exp) method.

-And if you want to see all experiments in the list you can use the listAll() method and if you see all experiments in a specific day, you can use listExp(day) method.

-For the sorting the list according to accuracy, you can use orderExperiment() method and if you sort experiments in a specific day you can use orderDay() method.

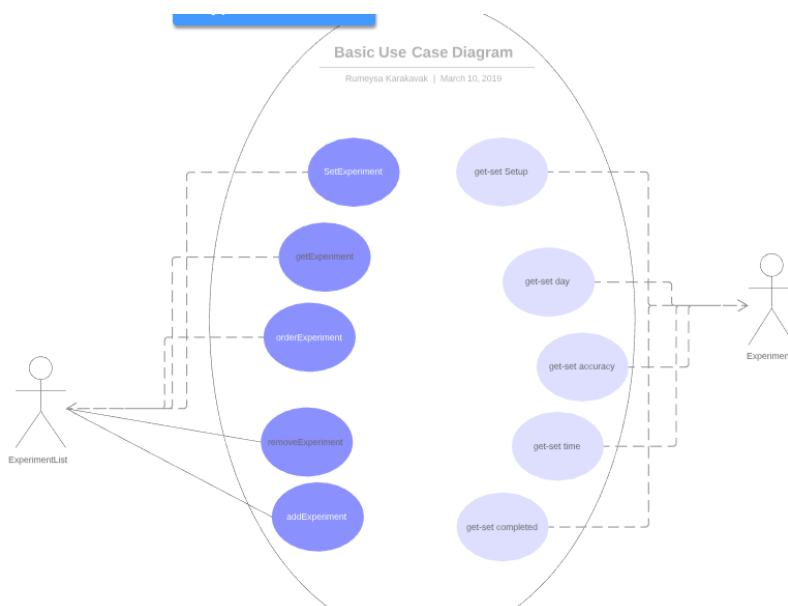
2 METHOD

2.1 Class Diagrams



There are 5 class -> Main, Experiment, Node, Experimentlist and ListIterator classes. ExperimentList class implements the Iterable. And ListIterator class implements the Iterator Class.

2.2 Use Case Diagrams



Experiment can get and set the setup, get and set the accuracy, get and set the time, get and set the day, get and set the is completed.

Experiment list can add new Experiment, remove an experiment specific, can order list according to accuracy.

2.3 Other Diagrams (optional)

Add other diagrams if required.

2.4 Problem Solution Approach

Keeping the experiments provided by own single LinkedList. So we can manage all experiments owing to ExperimentList.

3 RESULT

3.1 Test Cases

Possible scenarios:

```
for(int i=0; i<20; i++)
{
    System.out.println("-----");
    day = generator.nextInt(4) ;
    String setup = "setup"+Integer.toString(i);
    acc = (float) (i*0.1);
    Experiment e = new Experiment(setup, time , completed, day, acc);
    System.out.println("Add new experiment:");
    System.out.println(e.toString());
    list.addExp(e);
    list.listAll();
}

System.out.println("-----");
System.out.println("getExp(0,3) Result:");
Experiment e = list.getExp(0,3);
System.out.println(e.toString());
System.out.println("-----");
System.out.println("setExp(0,3), accuracy=1.0");
e.setAccuracy((float) 1.0);
list.setExp(0,3, e);
e = list.getExp(0,3);
System.out.println("-----");
System.out.println("getExp Result:");
e = list.getExp(0,3);
System.out.println(e.toString());
System.out.println("-----");
System.out.println("listExp(0) Result:");
list.listExp(0);
System.out.println("-----");
System.out.println("removeExp(0,0) Result:");
```

```

list.removeExp(0, 0);
list.listAll();
System.out.println("-----");
System.out.println("removeExp(1,0) Result:");
list.removeExp(1, 0);
list.listAll();
System.out.println("-----");
System.out.println("removeExp(1,) Result:");
list.removeExp(1, 0);
list.listAll();
System.out.println("-----");
System.out.println("removeExp(3,6) Result:");
list.removeExp(3, 6);
list.listAll();
System.out.println("-----");
System.out.println("orderExperiment Result:");
ExperimentList orderedList = list.orderExperiments();

Iterator itr = orderedList.iterator();
while(itr.hasNext()) {
    System.out.println(itr.next().toString());
}

```

```

Experiment exp1 = new Experiment("Exp1", 1, "04:11:1996", true, 18.0f);
Experiment exp2 = new Experiment("Exp2", 2, "04:11:1996", true, 11.0f);
Experiment exp3 = new Experiment("Exp3", 3, "04:11:1996", true, 19.0f);
Experiment exp4 = new Experiment("Exp4", 1, "04:11:1996", true, 9.0f);
Experiment exp5 = new Experiment("Exp5", 2, "04:11:1996", true, 8.0f);
Experiment exp6 = new Experiment("Exp6", 3, "04:11:1996", true, 17.0f);
Experiment exp7 = new Experiment("Exp7", 1, "04:11:1996", true, 15.0f);
ExperimentList expList = new ExperimentList();
expList.addExp(exp1);
expList.addExp(exp2);
expList.addExp(exp3);
expList.addExp(exp4);
expList.addExp(exp5);
expList.addExp(exp6);
expList.addExp(exp7);
System.out.println(expList.toString() + "\n\n");

expList.setExp(3,1, exp3);
System.out.println(expList.getExp(2,1).toString() + "\n\n");

System.out.println(expList.toString() + "\n\n");
expList.orderDay(1);
expList.orderDay(2);
expList.orderDay(3);
expList.orderExperiments();
expList.removeDay(1);
System.out.println(expList.toString() + "\n\n");
expList.removeDay(2);
System.out.println(expList.toString() + "\n\n");
expList.removeDay(3);
System.out.println(expList.toString() + "\n\n");

```

3.2 Running Results

```
Run exe
Experiment{setup='setup3', day=3, time='timeInfo', accuracy=0.3, completed=true}
-----
orderExperiment Result:
All experiment in linked list are :
Experiment{setup='setup1', day=2, time='timeInfo', accuracy=0.1, completed=true}
Experiment{setup='setup2', day=0, time='timeInfo', accuracy=0.2, completed=true}
Experiment{setup='setup3', day=3, time='timeInfo', accuracy=0.3, completed=true}
Experiment{setup='setup5', day=2, time='timeInfo', accuracy=0.5, completed=true}
Experiment{setup='setup6', day=3, time='timeInfo', accuracy=0.6, completed=true}
Experiment{setup='setup7', day=2, time='timeInfo', accuracy=0.7, completed=true}
Experiment{setup='setup8', day=0, time='timeInfo', accuracy=0.8, completed=true}
Experiment{setup='setup13', day=0, time='timeInfo', accuracy=1.0, completed=true}
Experiment{setup='setup10', day=2, time='timeInfo', accuracy=1.0, completed=true}
Experiment{setup='setup11', day=1, time='timeInfo', accuracy=1.1, completed=true}
Experiment{setup='setup12', day=3, time='timeInfo', accuracy=1.2, completed=true}
Experiment{setup='setup14', day=0, time='timeInfo', accuracy=1.4, completed=true}
Experiment{setup='setup16', day=3, time='timeInfo', accuracy=1.6, completed=true}
Experiment{setup='setup17', day=0, time='timeInfo', accuracy=1.7, completed=true}
Experiment{setup='setup18', day=3, time='timeInfo', accuracy=1.8, completed=true}
Experiment{setup='setup19', day=3, time='timeInfo', accuracy=1.9, completed=true}

Experiment{setup='setup19', day=3, time='timeInfo', accuracy=1.9, completed=true}

Process finished with exit code 0
```

```
Experiment{setup='setup9', day=3, time='timeInfo', accuracy=0.9, completed=true}
List day view:
Experiment{setup='setup2', day=0, time='timeInfo', accuracy=0.2, completed=true}
Experiment{setup='setup11', day=1, time='timeInfo', accuracy=1.1, completed=true}
Experiment{setup='setup0', day=2, time='timeInfo', accuracy=0.0, completed=true}
Experiment{setup='setup3', day=3, time='timeInfo', accuracy=0.3, completed=true}
-----
getExp(0,3) Result:
Experiment{setup='setup13', day=0, time='timeInfo', accuracy=1.3, completed=true}
Experiment{setup='setup13', day=0, time='timeInfo', accuracy=1.3, completed=true}
-----
setExp(0,3), accuracy=1.0
Experiment{setup='setup13', day=0, time='timeInfo', accuracy=1.0, completed=true}
-----
getExp Result:
Experiment{setup='setup13', day=0, time='timeInfo', accuracy=1.0, completed=true}
Experiment{setup='setup13', day=0, time='timeInfo', accuracy=1.0, completed=true}
-----
listExp(0) Result:
Experiment{setup='setup2', day=0, time='timeInfo', accuracy=0.2, completed=true}
Experiment{setup='setup4', day=0, time='timeInfo', accuracy=0.4, completed=true}
Experiment{setup='setup8', day=0, time='timeInfo', accuracy=0.8, completed=true}
Experiment{setup='setup13', day=0, time='timeInfo', accuracy=1.0, completed=true}
Experiment{setup='setup14', day=0, time='timeInfo', accuracy=1.4, completed=true}
Experiment{setup='setup17', day=0, time='timeInfo', accuracy=1.7, completed=true}
```

```
Add new experiment:
Experiment{setup='setup4', day=0, time='timeInfo', accuracy=0.4, completed=true}
List experiment view:
Experiment{setup='setup2', day=0, time='timeInfo', accuracy=0.2, completed=true}
Experiment{setup='setup4', day=0, time='timeInfo', accuracy=0.4, completed=true}
Experiment{setup='setup0', day=2, time='timeInfo', accuracy=0.0, completed=true}
Experiment{setup='setup1', day=2, time='timeInfo', accuracy=0.1, completed=true}
Experiment{setup='setup3', day=3, time='timeInfo', accuracy=0.3, completed=true}
List day view:
Experiment{setup='setup2', day=0, time='timeInfo', accuracy=0.2, completed=true}
Experiment{setup='setup0', day=2, time='timeInfo', accuracy=0.0, completed=true}
Experiment{setup='setup3', day=3, time='timeInfo', accuracy=0.3, completed=true}
-----
```

- Main titles -> 16pt , 2 line break
- Subtitles -> 14pt, 1.5 line break
- Paragraph -> 12pt, 1.5 line break