

**Gebze Technical University
Computer Engineering**

CSE 222 - 2019 Spring

HOMEWORK 2 REPORT

**MUHAMMET BURAK ÖZÇELİK
151044050**

Course Assistant: Ayşe Şerbetçi Turan

1 INTRODUCTION

1.1 Problem Definition

In this program, we will create an ExperimentList class to keep track of some machine learning experiments and their results.

In this structure, the day of experiments, accuracy, setup, time and completion information will be found. User can add and delete days. He can update his experiment. List the finished experiments. Sort all days or all experiments according to their accuracy.

You will find a detailed description of these in this document.

1.2 System Requirements

We will develop this program for computers.

Simply, we can partition this program into main two parts.

1. Experiment Side
 - a. Experiment Information
2. ExperimentList Side
 - a. Node
 - b. Iterator

Experiment Side

That is the part where the experiments will be held and processed.

Firstly, user must enter experiment information like experiment's day, accuracy, time, etc. into system to create experiment record.

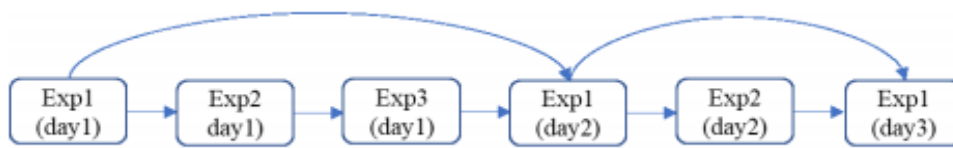
ExperimentList

ExperimentList experiments are a structure that connects each other with Node. Each experiment is held with a Node.

What can users do?

- Add experiments..
- It can erase one day's experiment or all experiments of that day.
- Can see or change the experiment of a day.
- List the completed experiments of a day.
- Can list the experiments of a day or all experiments according to their accuracy.

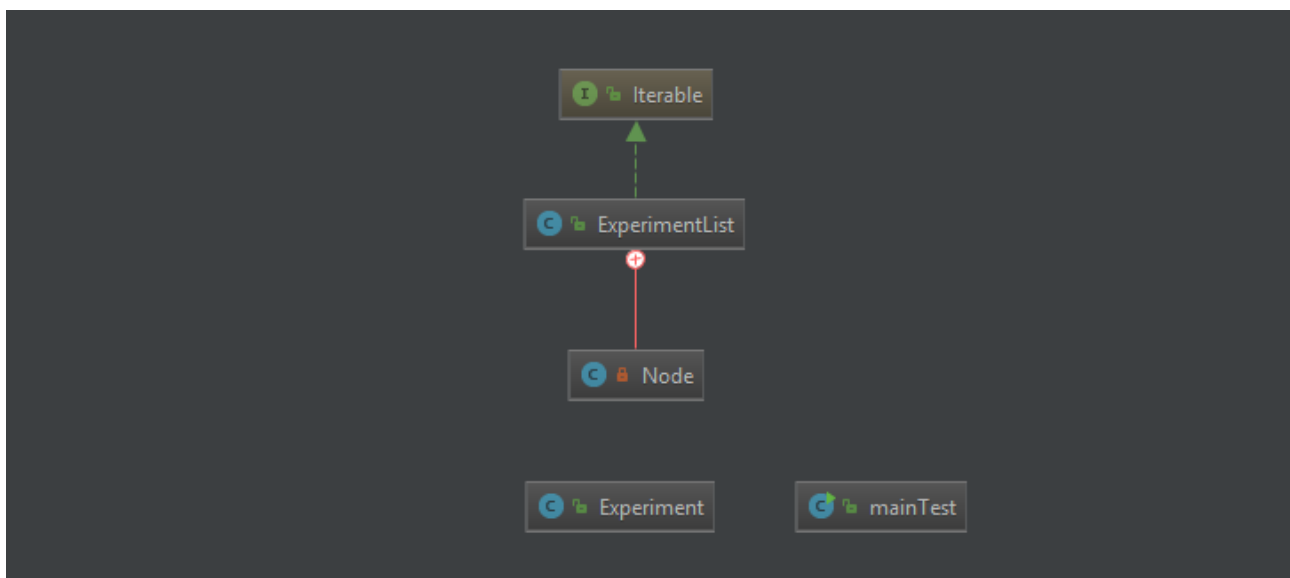
Example is as follows:



2 METHOD

2.1 Class Diagrams

- The ExperimentList class implements methods of Iterable.
- Node, private inner class. These ExperimentList classes are used to create a LinkedList Structure in mainTest.
- Experiment class is kept experiment's information.
- mainTest is the test class.



You can see detailed class diagrams below.

I	Iterable
m	iterator() <code>Iterator<T></code>
m	forEach(Consumer<? super T>) <code>void</code>
m	spliterator() <code>Spliterator<T></code>

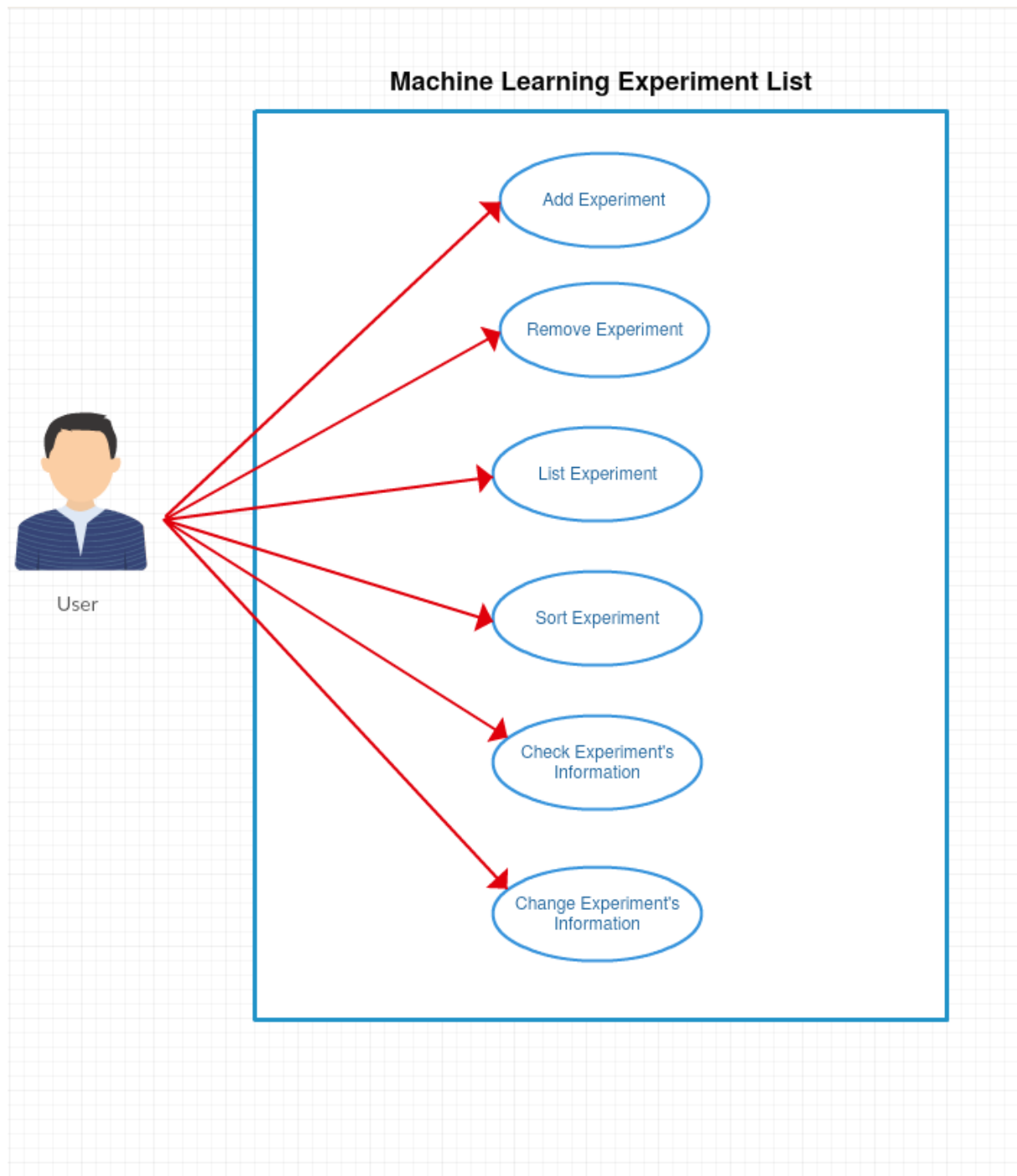
C	ExperimentList
m	ExperimentList()
m	size() <code>int</code>
m	iterator() <code>Iterator<Experiment></code>
m	addExp(Experiment) <code>void</code>
m	linkDay() <code>void</code>
m	getExp(int, int) <code>Experiment</code>
m	setExp(int, int, Experiment) <code>void</code>
m	removeExp(int, int) <code>void</code>
m	listExp(int) <code>void</code>
m	removeDay(int) <code>void</code>
m	orderDay(int) <code>void</code>
m	orderExperiments() <code>Node</code>
m	swap(Experiment, Experiment) <code>void</code>
m	print() <code>void</code>
p	head <code>Node</code>

C	Node
m	Node()
m	Node(Experiment)

C	mainTest
m	main(String[]) <code>void</code>

C	Experiment
m	Experiment(int, String, String, boolean, float)
m	Experiment(int, String, boolean, float)
m	Experiment()
m	toString() <code>String</code>
p	day <code>int</code>
p	completed <code>boolean</code>
p	accuracy <code>float</code>
p	setup <code>String</code>
p	time <code>String</code>

2.2 Use Case Diagrams



2.3 Other Diagrams (optional)

-

2.4 Problem Solution Approach

I will give you problem solution in a couple of parts

Creating Experiment

Firstly, the user should enter the information of experiment. (Day, setup, time etc).

Day: The day of experiment's start and integer (0,1,2...) that will use this program.

Setup: Explains the experimental setup. (String)

Time: String that experiment's time of start.

Accuracy: Output of experiment's success. (Float)

Completed: The information of experiment's that completed or not. (Boolean)

Create ExperimentList

User can add this experiment to the list or, if the list is full, it can be changed with another experiment's information by entering the appropriate day and index information.

ExperimentList Details

User can remove or check any experiment in the list that experiment's information by entering the appropriate day and index information.

The user can sort all the experiments in the list or the tests of the day according to their accuracy.

In this program

- User:
 - 1- Add Experiment: Add the created days to the list.
 - 2- Remove Experiment: Remove all the experiments for that day, or remove the day in the index you want that day.
 - 3- Get Experiment: Check the experiment information on the day you want.
 - 4- Set Experiment: Change the experiment information on the day you want.
 - 5- Order Day: Sort the experiments on the day you want according to accuracy.
 - 6- Order Experiment: : Sort all experiments that you want according to accuracy.

3 RESULT

3.1 Test Cases

Test Case No	Title	Input	Expected Output	Test Data	Real Output	Status
1	Add experiment to list	Experiment Information	List all elements after adding process.	(For Ex): "Expr1", dateFormat.format(date), true, 2, (float)0.1	On the console.	successful
2	Get an Experiment from List	Day - Index	Show the experiment that wanted.	<pre>Experiment e = container.getExp(2, 3);</pre>	On the console.	successful
3	Set an Experiment to List	Day – Index - Experiment	The changed information is displayed.	<pre>e.setAccuracy((float) 1.0); container.setExp(2, 3, e);</pre>	Accuracy changed.	successful
4	Remove an experiment from list with day an index	Day - Index	The list and the removed experiment are shown.	<pre>container.removeExp(0, 0);</pre>	On the console.	successful
5	Remove experiments from list with only day	Day	List prints.	<pre>container.removeExp(1);</pre>	On the console.	successful
6	Sort the list according to accuracy with day	Day	List prints	<pre>container.orderDay(2);</pre>	On the console.	successful
7	Sort all list according to accuracy	Day	List prints.	<pre>ExperimentList orderedList = container.orderExperiments();</pre>	On the console.	successful
8	List Experiment according to completion.	Day	List prints.	<pre>container.listExp(2);</pre>	On the console.	successful

3.2 Running Results

- Adding Experiment

We add an experiment and print List to show success.

```
-----
Add new experiment:
Experiment{setup='Expr1',   day=2,   time='08:51:06',   accuracy=0.1,   completed=true,}
List experiment view:
Experiment{setup='Expr1',   day=2,   time='08:51:06',   accuracy=0.1,   completed=true,}
List day view:
Experiment{setup='Expr1',   day=2,   time='08:51:06',   accuracy=0.1,   completed=true,}
-----
```

Figure 1

```
-----
Add new experiment:
Experiment{setup='Expr2',   day=1,   time='08:51:06',   accuracy=0.25,   completed=false,}
List experiment view:
Experiment{setup='Expr2',   day=1,   time='08:51:06',   accuracy=0.25,   completed=false,}
Experiment{setup='Expr1',   day=2,   time='08:51:06',   accuracy=0.1,   completed=true,}
List day view:
Experiment{setup='Expr2',   day=1,   time='08:51:06',   accuracy=0.25,   completed=false,}
Experiment{setup='Expr1',   day=2,   time='08:51:06',   accuracy=0.1,   completed=true,}
-----
```

Figure 2

Last state of Adding Process :

```
-----
Add new experiment:
Experiment{setup='Expr12',  day=3,   time='08:51:06',   accuracy=5.8,   completed=false,}
List experiment view:
Experiment{setup='Expr3',   day=0,   time='08:51:06',   accuracy=1.0,   completed=true,}
Experiment{setup='Expr4',   day=0,   time='08:51:06',   accuracy=0.1,   completed=false,}
Experiment{setup='Expr5',   day=0,   time='08:51:06',   accuracy=0.2,   completed=true,}
Experiment{setup='Expr6',   day=0,   time='08:51:06',   accuracy=0.3,   completed=false,}
Experiment{setup='Expr2',   day=1,   time='08:51:06',   accuracy=0.25,   completed=false,}
Experiment{setup='Expr1',   day=2,   time='08:51:06',   accuracy=0.1,   completed=true,}
Experiment{setup='Expr7',   day=2,   time='08:51:06',   accuracy=0.3,   completed=false,}
Experiment{setup='Expr8',   day=2,   time='08:51:06',   accuracy=7.4,   completed=true,}
Experiment{setup='Expr11',  day=2,   time='08:51:06',   accuracy=10.7,  completed=false,}
Experiment{setup='Expr10',  day=3,   time='08:51:06',   accuracy=0.6,   completed=true,}
Experiment{setup='Expr12',  day=3,   time='08:51:06',   accuracy=5.8,   completed=false,}
Experiment{setup='Expr9',   day=4,   time='08:51:06',   accuracy=0.5,   completed=false,}
List day view:
Experiment{setup='Expr3',   day=0,   time='08:51:06',   accuracy=1.0,   completed=true,}
Experiment{setup='Expr2',   day=1,   time='08:51:06',   accuracy=0.25,   completed=false,}
Experiment{setup='Expr1',   day=2,   time='08:51:06',   accuracy=0.1,   completed=true,}
Experiment{setup='Expr10',  day=3,   time='08:51:06',   accuracy=0.6,   completed=true,}
Experiment{setup='Expr9',   day=4,   time='08:51:06',   accuracy=0.5,   completed=false,}
-----
```

You can see list and next day node on the screen.

- Get / Set Experiment

The index and the days of the experiment change are shown.

```
-----
getExp(2,3) Result:
Experiment{setup='Expr11', day=2, time='08:51:06', accuracy=10.7, completed=false,}
-----
setExp(2,3), accuracy=1.0
-----
getExp Result:
Experiment{setup='Expr11', day=2, time='08:51:06', accuracy=1.0, completed=false,}
-----
```

- Remove Experiment

Removing processes Experiment.

Remove(day,index);

```
-----
removeExp(0,0) Result:
Experiment{setup='Expr3', day=0, time='09:03:22', accuracy=1.0, completed=true,}
List experiment view:
Experiment{setup='Expr4', day=0, time='09:03:22', accuracy=0.1, completed=false,}
Experiment{setup='Expr5', day=0, time='09:03:22', accuracy=0.2, completed=true,}
Experiment{setup='Expr6', day=0, time='09:03:22', accuracy=0.3, completed=false,}
Experiment{setup='Expr2', day=1, time='09:03:22', accuracy=0.25, completed=false,}
Experiment{setup='Expr1', day=2, time='09:03:22', accuracy=0.1, completed=true,}
Experiment{setup='Expr7', day=2, time='09:03:22', accuracy=0.3, completed=false,}
Experiment{setup='Expr8', day=2, time='09:03:22', accuracy=7.4, completed=true,}
Experiment{setup='Expr11', day=2, time='09:03:22', accuracy=1.0, completed=false,}
Experiment{setup='Expr10', day=3, time='09:03:22', accuracy=0.6, completed=true,}
Experiment{setup='Expr12', day=3, time='09:03:22', accuracy=5.8, completed=false,}
Experiment{setup='Expr9', day=4, time='09:03:22', accuracy=0.5, completed=false,}
List day view:
Experiment{setup='Expr4', day=0, time='09:03:22', accuracy=0.1, completed=false,}
Experiment{setup='Expr2', day=1, time='09:03:22', accuracy=0.25, completed=false,}
Experiment{setup='Expr1', day=2, time='09:03:22', accuracy=0.1, completed=true,}
Experiment{setup='Expr10', day=3, time='09:03:22', accuracy=0.6, completed=true,}
Experiment{setup='Expr9', day=4, time='09:03:22', accuracy=0.5, completed=false,}
-----
```

Remove(day);

```
-----
removeDay(1) Result:
Experiment{setup='Expr2', day=1, time='09:05:25', accuracy=0.25, completed=false,}
List experiment view:
Experiment{setup='Expr4', day=0, time='09:05:25', accuracy=0.1, completed=false,}
Experiment{setup='Expr5', day=0, time='09:05:25', accuracy=0.2, completed=true,}
Experiment{setup='Expr6', day=0, time='09:05:25', accuracy=0.3, completed=false,}
Experiment{setup='Expr1', day=2, time='09:05:25', accuracy=0.1, completed=true,}
Experiment{setup='Expr7', day=2, time='09:05:25', accuracy=0.3, completed=false,}
Experiment{setup='Expr8', day=2, time='09:05:25', accuracy=7.4, completed=true,}
Experiment{setup='Expr11', day=2, time='09:05:25', accuracy=1.0, completed=false,}
Experiment{setup='Expr10', day=3, time='09:05:25', accuracy=0.6, completed=true,}
Experiment{setup='Expr12', day=3, time='09:05:25', accuracy=5.8, completed=false,}
Experiment{setup='Expr9', day=4, time='09:05:25', accuracy=0.5, completed=false,}
List day view:
Experiment{setup='Expr4', day=0, time='09:05:25', accuracy=0.1, completed=false,}
Experiment{setup='Expr1', day=2, time='09:05:25', accuracy=0.1, completed=true,}
Experiment{setup='Expr10', day=3, time='09:05:25', accuracy=0.6, completed=true,}
Experiment{setup='Expr9', day=4, time='09:05:25', accuracy=0.5, completed=false,}
-----
```

- Order Day Experiment

```
orderDay(2) Result:
Experiment{setup='Expr1',    day=2,    time='09:07:37',    accuracy=0.1,    completed=true,}
Experiment{setup='Expr7',    day=2,    time='09:07:37',    accuracy=0.3,    completed=false,}
Experiment{setup='Expr11',   day=2,    time='09:07:37',    accuracy=1.0,    completed=false,}
Experiment{setup='Expr8',    day=2,    time='09:07:37',    accuracy=7.4,    completed=true,}
```

- Order All Experiment

```
orderExperiment Result:
Experiment{setup='Expr4',    day=0,    time='09:07:37',    accuracy=0.1,    completed=false,}
Experiment{setup='Expr1',    day=2,    time='09:07:37',    accuracy=0.1,    completed=true,}
Experiment{setup='Expr5',    day=0,    time='09:07:37',    accuracy=0.2,    completed=true,}
Experiment{setup='Expr6',    day=0,    time='09:07:37',    accuracy=0.3,    completed=false,}
Experiment{setup='Expr7',    day=2,    time='09:07:37',    accuracy=0.3,    completed=false,}
Experiment{setup='Expr9',    day=4,    time='09:07:37',    accuracy=0.5,    completed=false,}
Experiment{setup='Expr10',   day=3,    time='09:07:37',    accuracy=0.6,    completed=true,}
Experiment{setup='Expr11',   day=2,    time='09:07:37',    accuracy=1.0,    completed=false,}
Experiment{setup='Expr12',   day=3,    time='09:07:37',    accuracy=5.8,    completed=false,}
Experiment{setup='Expr8',    day=2,    time='09:07:37',    accuracy=7.4,    completed=true,}
```

3.3 Complexity of Methods

Method Name	Complexity	Details
AddExp	$\Theta(n)$	One loop and size of list. $\Theta((\Theta(n) + \Theta(n))) = \Theta(n)$
linkDay	$\Theta(n)$	One loop and size of list.
getExp	$\Theta(n)$	One loop and size of list.
setExp	$\Theta(n)$	One loop and size of list. $\Theta((\Theta(n) + \Theta(1))) = \Theta(n)$
removeExp	$\Theta(n)$	One loop and size of list. $\Theta((\Theta(n) + \Theta(1))) = \Theta(n)$
listExp	$\Theta(n)$	One loop and size of list.
removeDay	$\Theta(n^2)$	One loop and size of list and calling removeExp method. $\Theta((\Theta(n) * \Theta(n))) = \Theta(n^2)$
orderDay	$\Theta(n^2)$	Bubble sort algorithm $\Theta(n)$
orderExperiments	$\Theta(n^2)$	Bubble sort algorithm $\Theta(n)$
Swap	$\Theta(1)$ / constant	Change only datas, constant complexity.
listAll	$\Theta(n)$	One loop and size of list.