

**Gebze Technical University
Computer Engineering**

CSE 222 - 2018 Spring

HOMEWORK 4 REPORT

**BURHAN ELGÜN
151044060**

Course Assistant: MEHMET BURAK KOCA

1 INTRODUCTION

1.1 Problem Definition

PART-1

General Tree yi Binary Tree den extend ederek implement etmek.Bu ağaç türünde sol taraf çocukları sağ taraf kardeşleri temsil etmektedir ve bu ağaç türüne göre traverse methodları override edilmelidir.

PART-2

Nodun da bir boyuttan fazla eleman taşıyan bir tree impelement edilmesi gerekmekte ve bu tree de her bir levelde farklı dimesionlar birbiriyle karşılaştırılarak search tree gibi yerleştirilmelidir.Yani bu kurala göre add methodu implement edilmeli.Ayrıca search interfacesindeki methodlar implement edilmelidir ve BinaryTreeden extend olmalıdır.

1.2 System Requirements

PART-1

public boolean add(E parentItem,E childItem){ methodu bir parentItem ve childItem e ihtiyaç duyar bu sayede lokasyon belirlenmiş olur ve childItem eklenmiş olur

public Node postOrderSearch(E name, Node node){ bu method name e ihtiyaç duyar bu aranılan elemanın kendisidir ayrıca aranmasını istediği treenin root una da ihtiyaç duyar

public void postOrderTraverse(Node node){bu method postordertraverse işlemini yapmak istenen treenin root u na itiyaç duyar ve bu root parametre olarak gönderilir.

public void preOrderTraverse(Node node){ bu method preordertraverse işlemini yapmak istenen treenin root u na itiyaç duyar ve bu root parametre olarak gönderilir.

public Node levelOrderSearch(E name, Node node){){ bu method name e ihtiyaç duyar bu aranılan elemanın kendisidir ayrıca aranmasını istediği treenin root una da ihtiyaç duyar

public void levelOrderTraverse(Node node){ bu method levelordertraverse işlemini yapmak istenen treenin root u na itiyaç duyar ve bu root parametre olarak gönderilir.

PART-2

int dimensionSize; tree'nin kaç dimensionlu olduğunun tutulması gereken bir data field a ihtiyaç vardır.

public boolean add(E item){ eklenmek istenen item olan E tipinde bir parametreye ihtiyaç vardır

public boolean contains(E target){ var olup ı olmadığının sorgulandığı elemana parametre olarak ihtiyaç vardır

public E find(E target){ var olup ı olmadığının sorgulandığı elemana parametre olarak ihtiyaç vardır

public E delete(E target){silinmesi istenen elemanın parametre olarak gönderilmesine ihtiyaç vardır.

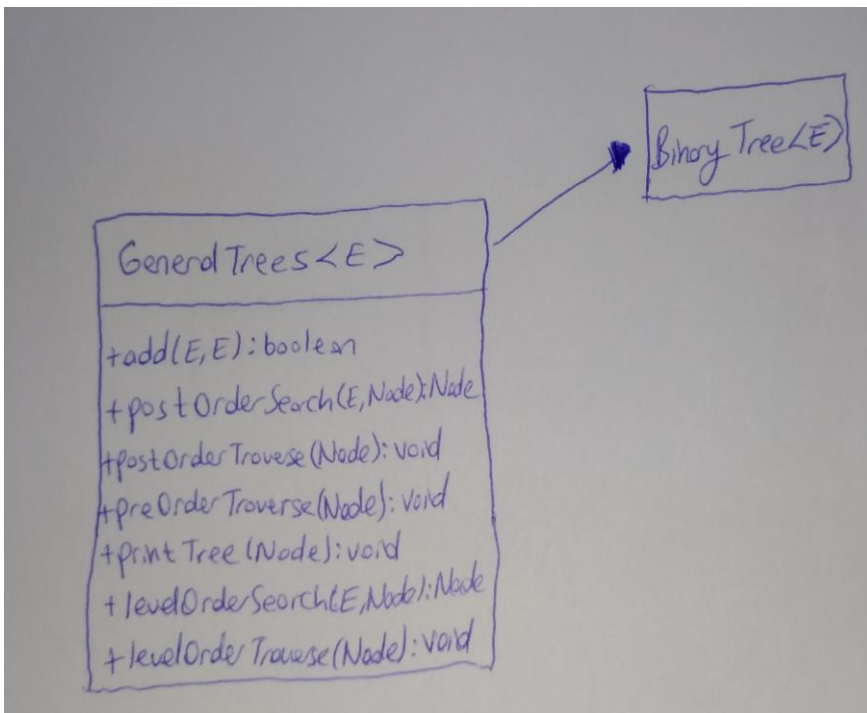
public boolean remove(E target){ silinmesi istenen elemanın parametre olarak gönderilmesine ihtiyaç vardır.

void printPreorder(Node node){ preordertraverse ederken print işlemini gerçekleştirmek için parametre olarak root ihtiyacı vardır.

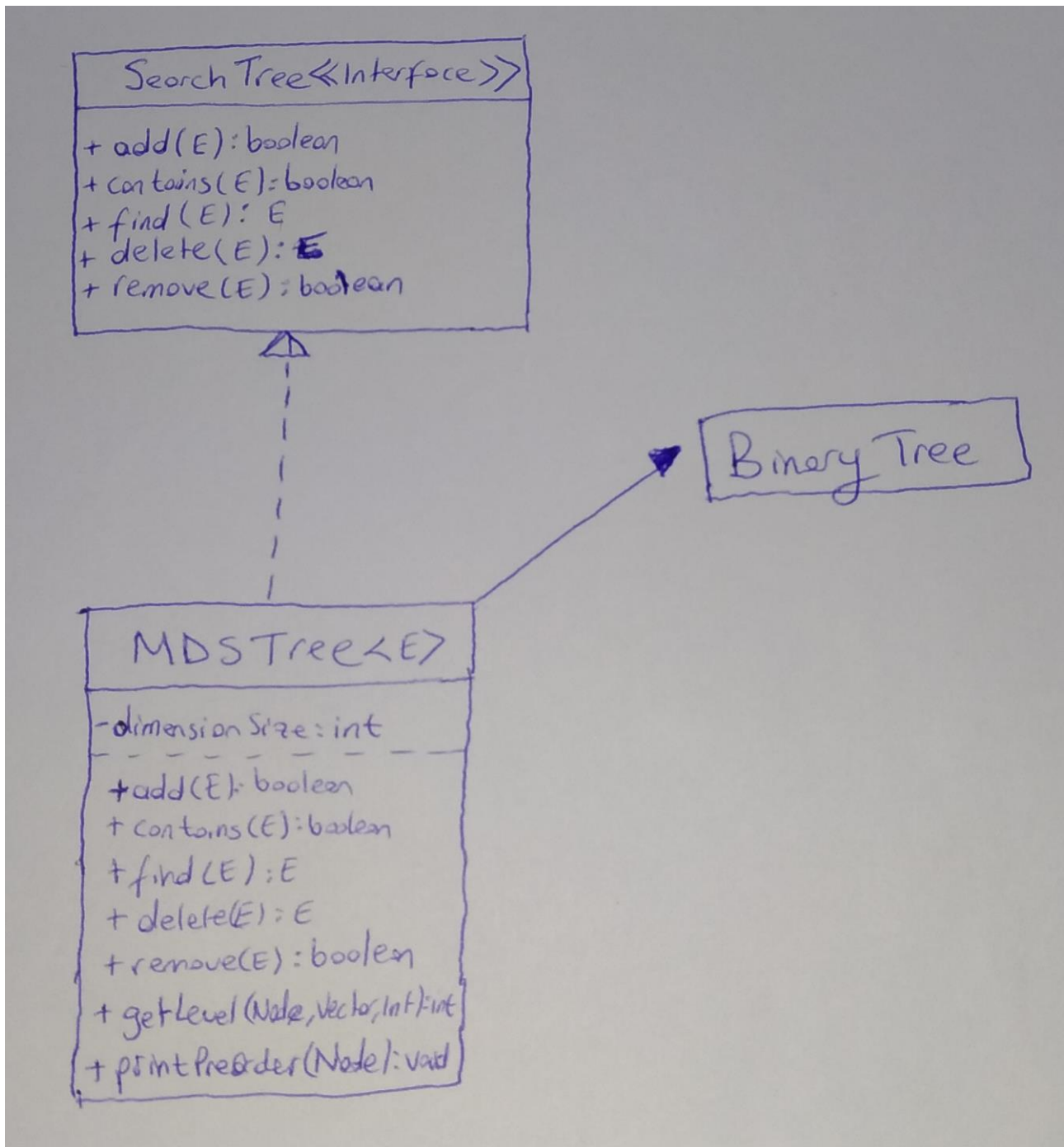
2 METHOD

2.1 Class Diagrams

PART-1



PART-2



2.2 Use Case Diagrams

Add use case diagrams if required.

2.3 Other Diagrams (optional)

Add other diagrams if required.

2.4 Problem Solution Approach

PART-1

General treede elelman eklerken eğer verilen parentin çocuğu zaten varsa yeni çocuğu eski çocuğun sağına aldım bu sayede o çocukla kardeş olmuş oldular.(right sibling)

Level order traverse te bir nodun önce sağ çocuklarını(sibling) null olana kadar takip ettim ve nul geldiğinde diğer çocuğa gidip tekrar kardeşlerine bakmaya devam ettim.

Postorder traverse de ise en sol alttan itibaren parent e ve parentin kardeşlerine baktım.

Preorder travers aynı çıktı.

PART-2

DimesionSize 1 tuttum bu sayede eğer 3 dimensionlu bir tree gelirse yeni gelen elemanları eklerken her leveldeki elemanın levelinin 3 ile moduna göre ya x ya yada z ile karşılaştırırım.Her leveldeki elemanın levelini öğrenmek için ise item alıp node döndüren bir method yazdım.

3 RESULT

3.1 Test Cases

PART-1

Debian_GTU_svy - VMware Workstation 14 Player (Non-commercial use only)

hw4-part1 [-AdeapProjects/hw4-part1] - .../src/GeneralTreesTest.java [hw4-part1] - IntelliJ IDEA

File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help

hw4-part1 | src | GeneralTreesTest

Run GeneralTreesTest All 6 tests passed - 91ms

Test Results

- GeneralTreesTest (91ms)
 - postOrderSearch() (32ms)
 - postOrderTraverse() (13ms)
 - preOrderTraverse() (2ms)
 - levelOrderSearch() (10ms)
 - add() (32ms)
 - levelOrderTraverse() (2ms)

```

/nmr/lib/jvm/java-1.8.0-gensdk-1386/bin/java ...
*****POSTORDERSEARCH METHOD TESTING*****
FIRST TREE
parent: 1, child: 2 eklenmiştir
parent: 2, child: 3 eklenmiştir
parent: 3, child: 4 eklenmiştir
parent: 4, child: 5 eklenmiştir
parent: 5, child: 6 eklenmiştir
parent: 6, child: 7 eklenmiştir
parent: 7, child: 8 eklenmiştir
parent: 8, child: 9 eklenmiştir
parent: 9 child: 10 eklenmiştir
parent: 10, child: 11 eklenmiştir
parent: 11, child: 12 eklenmiştir
parent: 12, child: 13 eklenmiştir
parent: 13, child: 14 eklenmiştir

Tree 7 içerir mi (postOrderSearch kullanarak)
a.postOrderSearch(7,a.root) = 7
Tree 7 içerir

Tree -99 içerir mi (postOrderSearch kullanarak)
a.postOrderSearch(-99,a.root) = null
Tree -99 içermez
*****

SECOND TREE
parent: 14, child: 13 eklenmiştir
parent: 13, child: 12 eklenmiştir
parent: 12, child: 11 eklenmiştir
parent: 11, child: 10 eklenmiştir
parent: 10, child: 9 eklenmiştir
parent: 9, child: 8 eklenmiştir
parent: 8, child: 7 eklenmiştir
parent: 7, child: 6 eklenmiştir
parent: 6, child: 5 eklenmiştir
parent: 5, child: 4 eklenmiştir
parent: 4, child: 3 eklenmiştir
parent: 3, child: 2 eklenmiştir

Tree 7 içerir mi (postOrderSearch kullanarak)
b.postOrderSearch(7,b.root) = 7
Tree 7 içerir

```

Tests Passed: 6 passed (moments ago)

hw4-part1 [-Ade...

8:32 Lf: UTF-8 22:06 16.04.2018

Debian_GTU_svy - VMware Workstation 14 Player (Non-commercial use only)

hw4-part1 [-AdeapProjects/hw4-part1] - .../src/GeneralTreesTest.java [hw4-part1] - IntelliJ IDEA

File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help

hw4-part1 | src | GeneralTreesTest

Run GeneralTreesTest All 6 tests passed - 91ms

Test Results

- GeneralTreesTest (91ms)
 - postOrderSearch() (32ms)
 - postOrderTraverse() (13ms)
 - preOrderTraverse() (2ms)
 - levelOrderSearch() (10ms)
 - add() (32ms)
 - levelOrderTraverse() (2ms)

```

b.postOrderSearch(-99,b.root) = null
Tree -99 içermez
*****
*****POSTORDERTRAVERSE METHOD TESTING*****
FIRST TREE
parent: 1, child: 2 eklenmiştir
parent: 2, child: 3 eklenmiştir
parent: 3, child: 4 eklenmiştir
parent: 4, child: 5 eklenmiştir
parent: 5, child: 6 eklenmiştir
parent: 6, child: 7 eklenmiştir
parent: 7, child: 8 eklenmiştir
parent: 8, child: 9 eklenmiştir
parent: 9 child: 10 eklenmiştir
parent: 10, child: 11 eklenmiştir
parent: 11, child: 12 eklenmiştir
parent: 12, child: 13 eklenmiştir
parent: 13, child: 14 eklenmiştir
post order traverse = 14 13 12 11 10 9 8 7 6 5 4 3 2 1
*****

SECOND TREE
parent: 14, child: 13 eklenmiştir
parent: 13, child: 12 eklenmiştir
parent: 12, child: 11 eklenmiştir
parent: 11, child: 10 eklenmiştir
parent: 10, child: 9 eklenmiştir
parent: 9, child: 8 eklenmiştir
parent: 8, child: 7 eklenmiştir
parent: 7, child: 6 eklenmiştir
parent: 6, child: 5 eklenmiştir
parent: 5, child: 4 eklenmiştir
parent: 4, child: 3 eklenmiştir
parent: 3, child: 2 eklenmiştir
post order traverse = 2 3 4 5 6 7 8 9 10 11 12 13 14
*****

*****PREORDERTRAVERSE METHOD TESTING*****
FIRST TREE

```

Tests Passed: 6 passed (a minute ago)

hw4-part1 [-Ade...

8:32 Lf: UTF-8 22:07 16.04.2018

Debian_GTU_svy - VMware Workstation 14 Player (Non-commercial use only)

hw4-part1 [-/IdeaProjects/hw4-part1] - .../src/GeneralTreesTest.java [hw4-part1] - IntelliJ IDEA

File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help

hw4-part1 src GeneralTreesTest

Run GeneralTreesTest All 6 tests passed - 91ms

Test Results

- GeneralTreesTest (91ms)
- postOrderSearch() (32ms)
- postOrderTraverse() (13ms)
- preOrderTraverse() (2ms)
- levelOrderSearch() (10ms)
- add() (32ms)
- levelOrderTraverse() (2ms)

```
*****LEVELORDERSEARCH METHOD TESTING*****  
FIRST TREE  
*****  
parent: 1, child: 2 eklenmiştir  
parent: 2, child: 3 eklenmiştir  
parent: 3, child: 4 eklenmiştir  
parent: 4, child: 5 eklenmiştir  
parent: 5, child: 6 eklenmiştir  
parent: 6, child: 7 eklenmiştir  
parent: 7, child: 8 eklenmiştir  
parent: 8, child: 9 eklenmiştir  
parent: 9 child: 10 eklenmiştir  
parent: 10, child: 11 eklenmiştir  
parent: 11, child: 12 eklenmiştir  
parent: 12, child: 13 eklenmiştir  
parent: 13, child: 14 eklenmiştir  
  
Tree 7 içerir mi (levelOrderSearch kullanarak)  
a.levelOrderSearch(7,a.root) = 7  
Tree 7 içerir  
  
Tree -99 içerir mi (levelOrderSearch kullanarak)  
a.levelOrderSearch(-99,a.root) = null  
Tree -99 içermez  
*****  
  
SECOND TREE  
*****  
parent: 14, child: 13 eklenmiştir  
parent: 13, child: 12 eklenmiştir  
parent: 12, child: 11 eklenmiştir  
parent: 11, child: 10 eklenmiştir  
parent: 10, child: 9 eklenmiştir  
parent: 9, child: 8 eklenmiştir  
parent: 8, child: 7 eklenmiştir  
parent: 7, child: 6 eklenmiştir  
parent: 6, child: 5 eklenmiştir  
parent: 5, child: 4 eklenmiştir  
parent: 4, child: 3 eklenmiştir  
parent: 3, child: 2 eklenmiştir  
  
Tree 7 içerir mi (levelOrderSearch kullanarak)  
b.levelOrderSearch(7,b.root) = 7  
Tree 7 içerir
```

Tests Passed: 6 passed (2 minutes ago)

8:32 LF: UTF-8: 22:08 16.04.2018

Debian_GTU_svy - VMware Workstation 14 Player (Non-commercial use only)

hw4-part1 [-/IdeaProjects/hw4-part1] - .../src/GeneralTreesTest.java [hw4-part1] - IntelliJ IDEA

File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help

hw4-part1 src GeneralTreesTest

Run GeneralTreesTest All 6 tests passed - 91ms

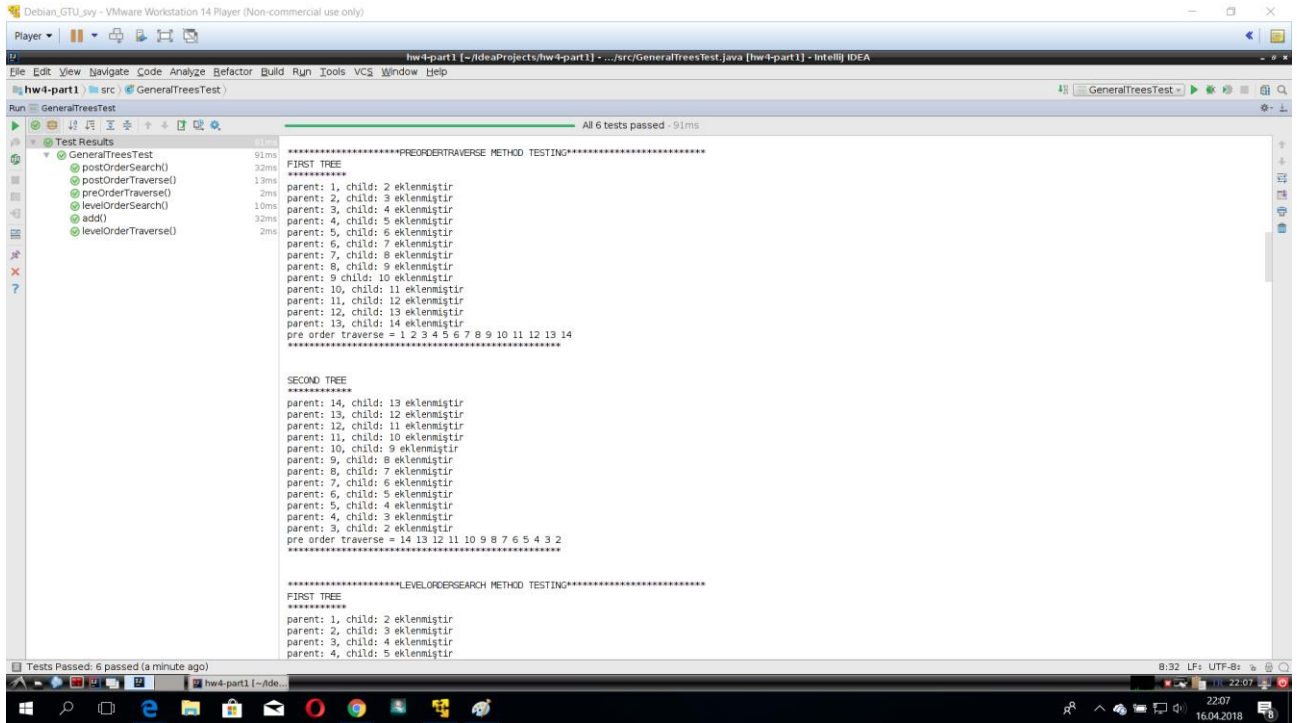
Test Results

- GeneralTreesTest (91ms)
- postOrderSearch() (32ms)
- postOrderTraverse() (13ms)
- preOrderTraverse() (2ms)
- levelOrderSearch() (10ms)
- add() (32ms)
- levelOrderTraverse() (2ms)

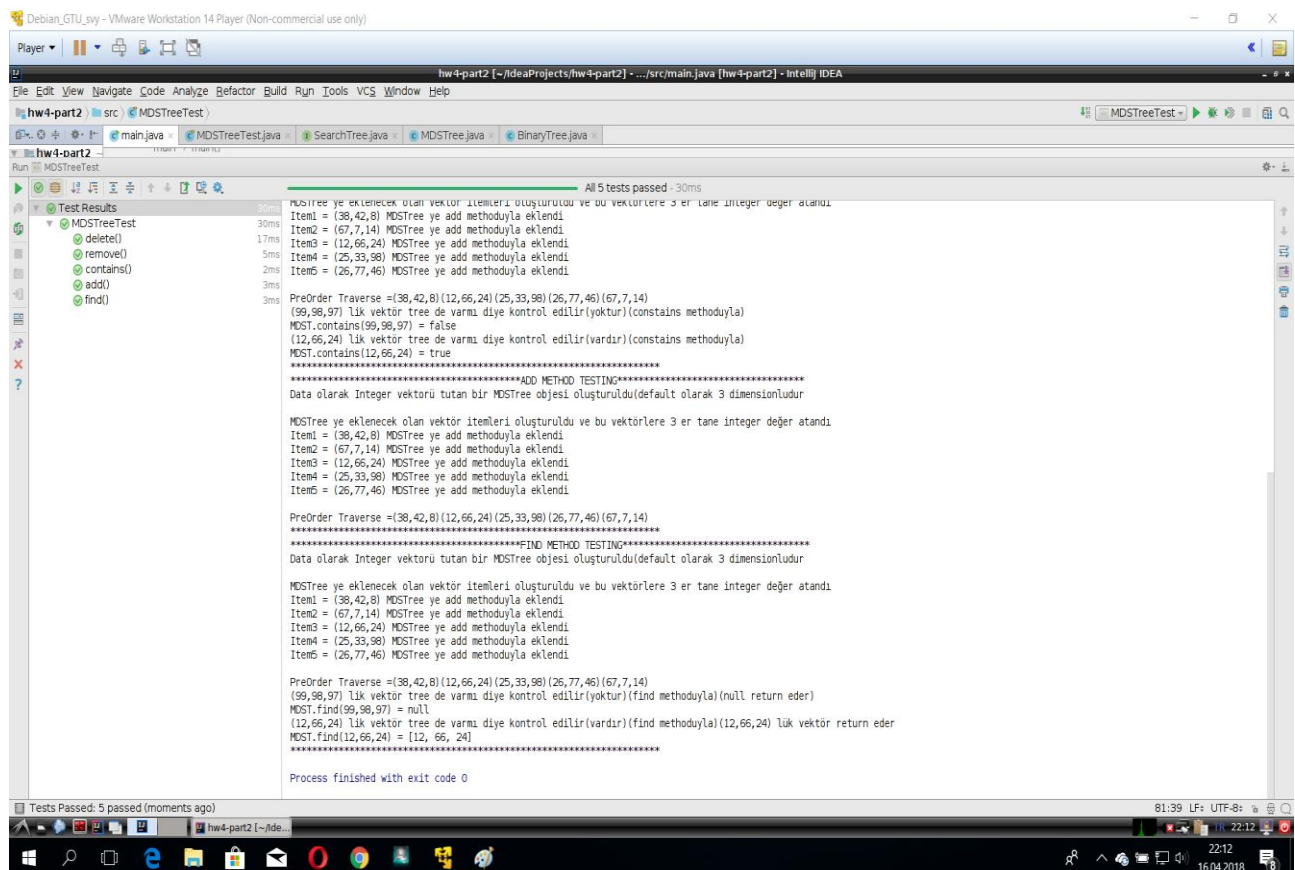
```
*****ADD METHOD TESTING*****  
FIRST TREE  
*****  
parent: 1, child: 2 eklenmiştir  
1  
2  
post order traverse = 2 1  
pre order traverse = 1 2  
*****  
  
parent: 1, child: 4 eklenmiştir  
1  
2  
4  
post order traverse = 2 4 1  
pre order traverse = 1 2 4  
*****  
  
parent: 1, child: 7 eklenmiştir  
1  
2  
4  
7  
post order traverse = 2 4 7 1  
pre order traverse = 1 2 4 7  
*****  
  
parent: 2, child: 3 eklenmiştir  
1  
2  
3 4  
7  
post order traverse = 3 2 4 7 1  
pre order traverse = 1 2 3 4 7  
*****  
  
parent: 2, child: 6 eklenmiştir  
1  
2  
3 4  
6 7  
post order traverse = 3 6 2 4 7 1  
pre order traverse = 1 2 3 6 4 7  
*****
```

Tests Passed: 6 passed (2 minutes ago)

8:32 LF: UTF-8: 22:08 16.04.2018

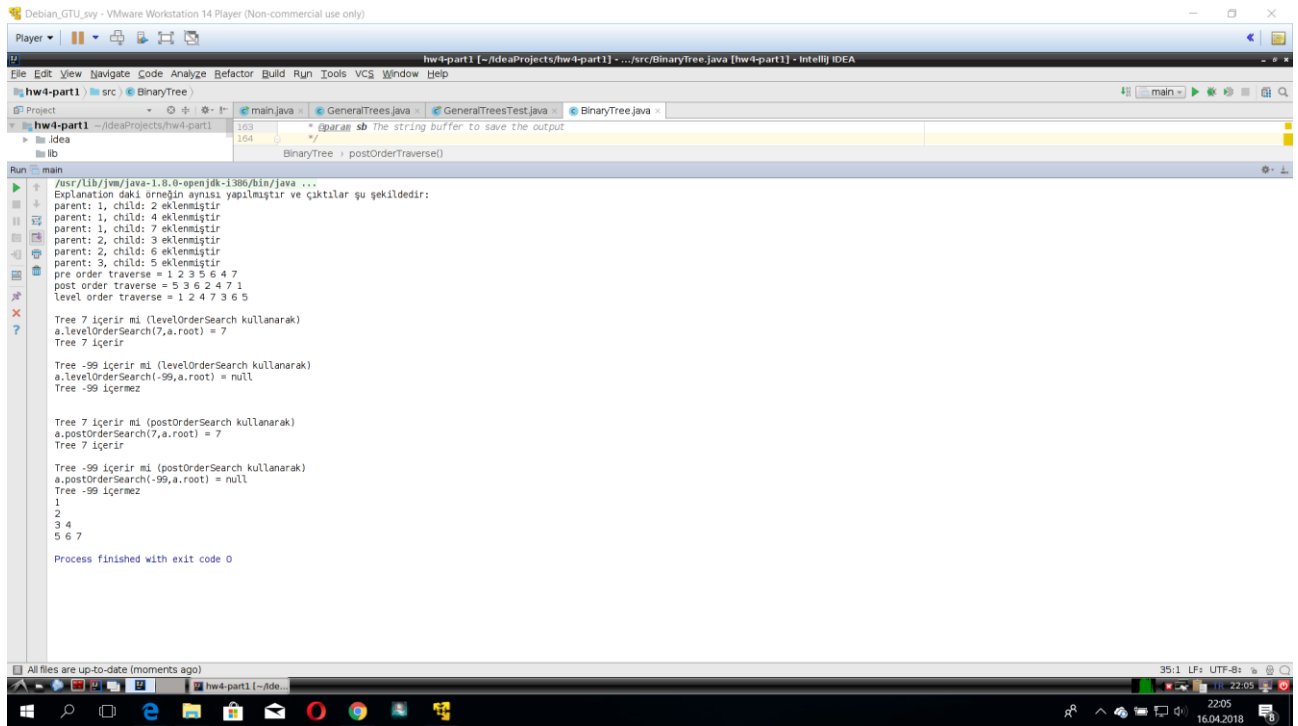


The screenshot shows the IntelliJ IDEA IDE with a Java project named 'hw4-part2'. The main editor displays the 'MDSTreeTest.java' file, which contains a series of test cases for an 'MDSTree' class. The tests are organized into sections: 'DELETE METHOD TESTING', 'REMOVE METHOD TESTING', and 'CONTAINS METHOD TESTING'. Each section includes a 'Data olarak Integer vektörü tutan bir MDSTree objesi oluşturuldu' (Data as Integer vector stored in an MDSTree object created) statement, followed by a list of items added to the tree, a 'LevelOrder Traverse' method call, and a 'kontrol etmek için levelOrder Traverse' (control for levelOrder Traverse) statement. The 'Test Results' window on the left shows that all 5 tests passed successfully. The bottom status bar indicates 'Tests Passed: 5 passed (moments ago)'.



3.2 Running Results

PART-1



```
Run - main
/usr/lib/jvm/java-1.8.0-openjdk-1.386/bin/java ...
Explanationdaki örneğin aynı yapılmıştır ve çıktılar şu şekildedir:
parent: 1, child: 2 eklenmiştir
parent: 1, child: 4 eklenmiştir
parent: 1, child: 7 eklenmiştir
parent: 2, child: 3 eklenmiştir
parent: 2, child: 6 eklenmiştir
parent: 3, child: 5 eklenmiştir
pre order traverse = 1 2 3 5 6 4 7
post order traverse = 5 3 6 2 4 7 1
level order traverse = 1 2 4 7 3 6 5

Tree 7 içerir mi (levelOrderSearch kullanarak)
a.levelOrderSearch(7,a.root) = 7
Tree 7 içerir

Tree -99 içerir mi (levelOrderSearch kullanarak)
a.levelOrderSearch(-99,a.root) = null
Tree -99 içermez

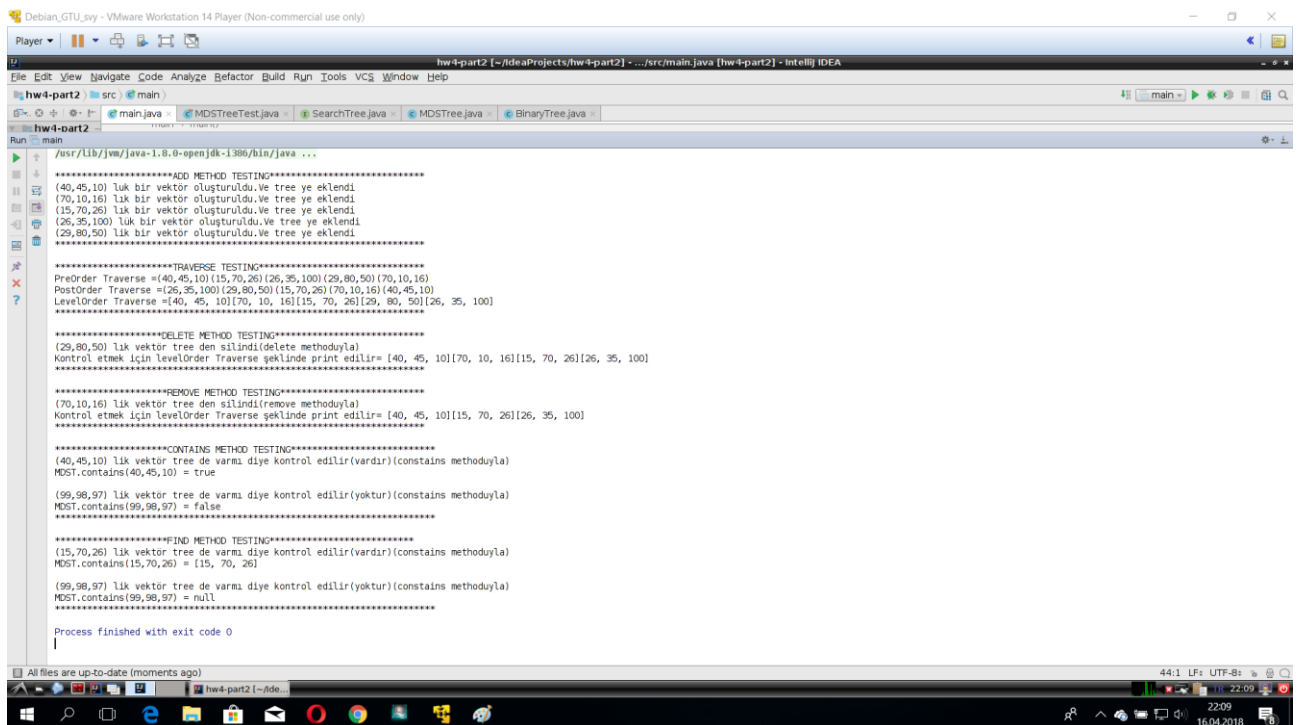
Tree 7 içerir mi (postOrderSearch kullanarak)
a.postOrderSearch(7,a.root) = 7
Tree 7 içerir

Tree -99 içerir mi (postOrderSearch kullanarak)
a.postOrderSearch(-99,a.root) = null
Tree -99 içermez

1
2
3 4
5 6 7

Process finished with exit code 0
```

PART-2



```
Run - main
/usr/lib/jvm/java-1.8.0-openjdk-1.386/bin/java ...
*****ADD METHOD TESTING*****
(40,45,10) luk bir vektör oluşturuldu.Ve tree ye eklendi
(70,10,16) luk bir vektör oluşturuldu.Ve tree ye eklendi
(15,70,26) luk bir vektör oluşturuldu.Ve tree ye eklendi
(26,35,100) luk bir vektör oluşturuldu.Ve tree ye eklendi
(29,80,50) luk bir vektör oluşturuldu.Ve tree ye eklendi
*****TRAVERSE TESTING*****
PreOrder Traverse =[40,45,10](15,70,26)[26,35,100](29,80,50)(70,10,16)
PostOrder Traverse =[26,35,100](29,80,50)(15,70,26)(70,10,16)(40,45,10)
LevelOrder Traverse =[40, 45, 10][70, 10, 16][15, 70, 26][29, 80, 50][26, 35, 100]
*****DELETE METHOD TESTING*****
(29,80,50) luk vektör tree den silindi(delete methoduyla)
Kontrol etmek için levelOrder Traverse şeklinde print edilir= [40, 45, 10][70, 10, 16][15, 70, 26][26, 35, 100]
*****REMOVE METHOD TESTING*****
(70,10,16) luk vektör tree den silindi(remove methoduyla)
Kontrol etmek için levelOrder Traverse şeklinde print edilir= [40, 45, 10][15, 70, 26][26, 35, 100]
*****CONTAINS METHOD TESTING*****
(40,45,10) luk vektör tree de varmı diye kontrol edilir(vardır)(contains methoduyla)
MDST.contains(40,45,10) = true
(99,98,97) luk vektör tree de varmı diye kontrol edilir(yoktur)(contains methoduyla)
MDST.contains(99,98,97) = false
*****FIND METHOD TESTING*****
(15,70,26) luk vektör tree de varmı diye kontrol edilir(vardır)(contains methoduyla)
MDST.contains(15,70,26) = [15, 70, 26]
(99,98,97) luk vektör tree de varmı diye kontrol edilir(yoktur)(contains methoduyla)
MDST.contains(99,98,97) = null

Process finished with exit code 0
```

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