

**Gebze Technical University
Computer Engineering**

CSE 222 - 2018 Spring

HOMEWORK 04 REPORT

**MESUT BUNALDI
111044077**

Course Assistant:
Fatma Nur ESİRCİ
Tuğbagül ALTAN AKIN
Mehmet Burak KOCA

1 INTRODUCTION

1.1 Problem Definition

1. Para :Veri yapıları kavramlarından birisi olan Tree yapısı üzerinde traversing, search ve add gibi işlemlerin yapılabilmesi için gerekli interface,class ve bunların extend / implement edilen field ve metotlarının kullanılarak yeni bir General Tree yapısı oluşturulması ve test edilmesi istenmiştir.

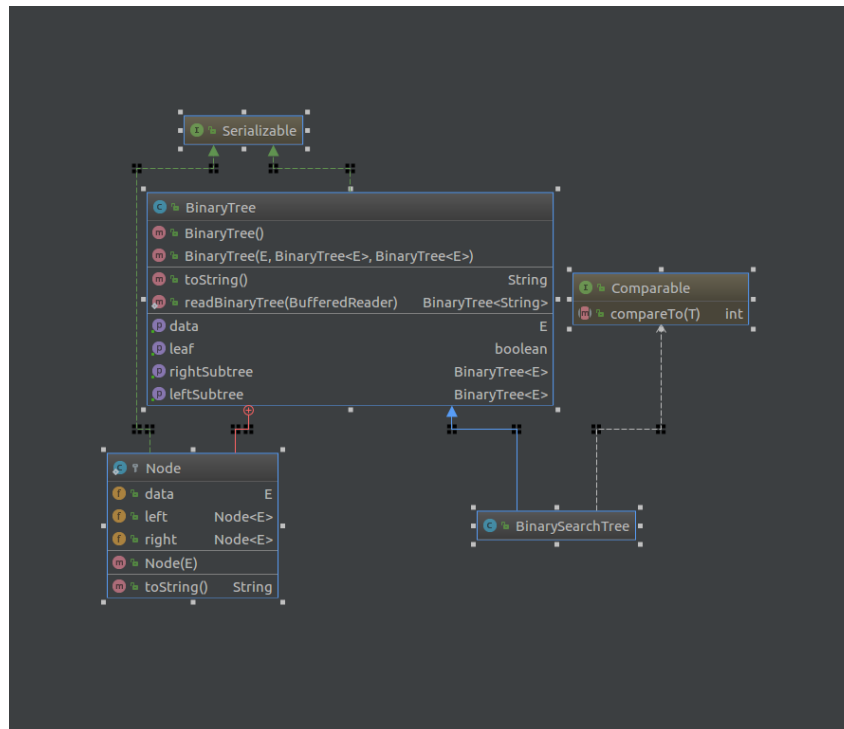
2. Para : İçerisinde çok boyutlu elementler bulundurabilme özelliğine sahip Search Tree yapısı oluşturulması istenmiştir. Bu yapı içerisinde çok boyutlu elementlerin her bir boyutunda yer alan verilerin bulunulan levellere göre birbiri ile karşılaştırılması ve traverse işleminin karşılaştırma sonucuna göre (add veya remove gibi işlemlerin) yapılması istenmiştir.

1.2 System Requirements

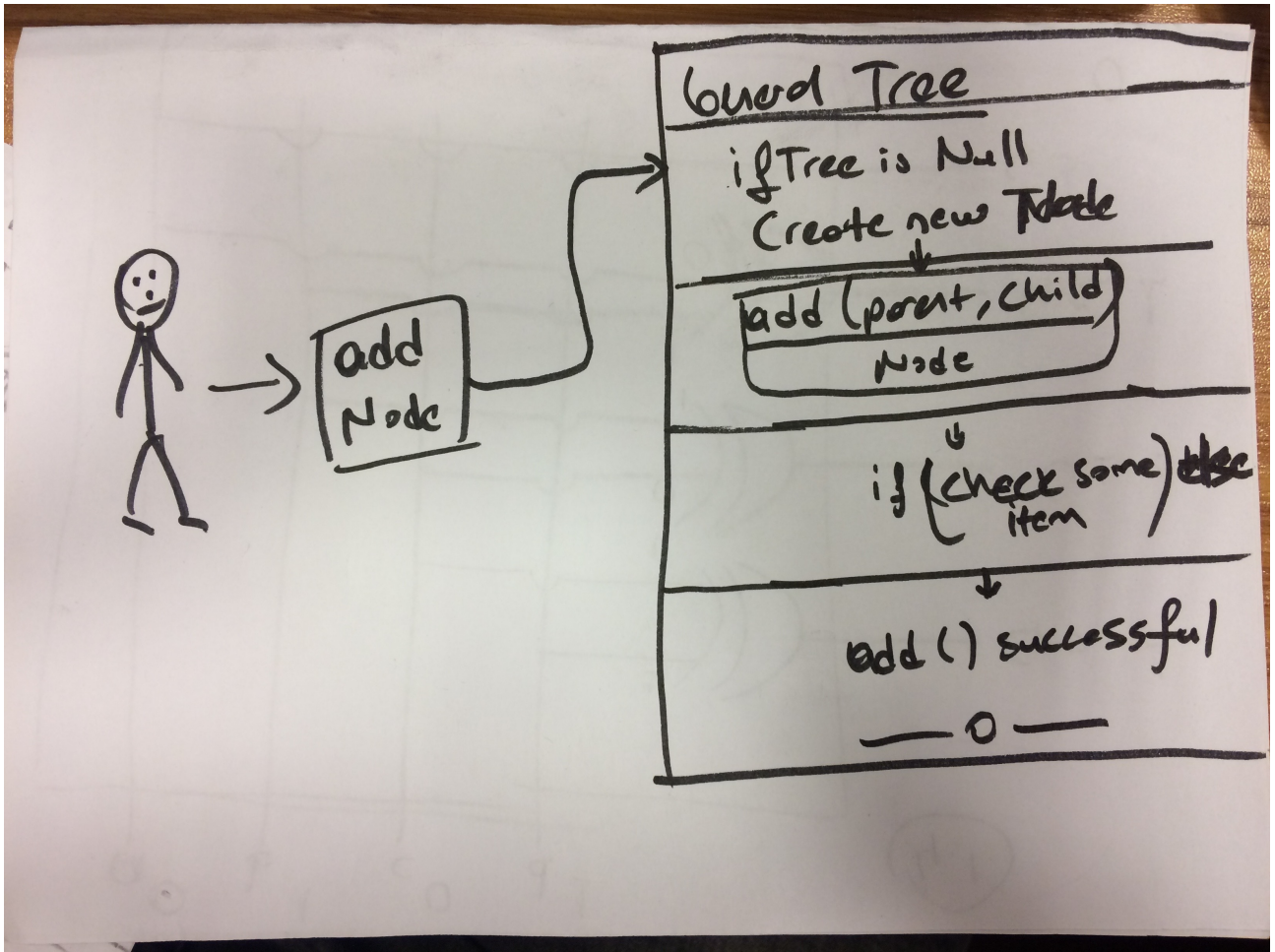
Proje içerisinde kullanılmak üzere General Tree sınıfını extend edeceğimiz generic bir Binary Tree classına ihtiyacımız vardır.(Bu class ders kitabından temin edilmiştir.) Ayrıca oluşturmuş olduğumuz General Tree yapısı içerisine yapıyı kullanabilmek için gerekli olan verilerin kullanıcı tarafından obje oluşturulduktan sonra eklenmesi gerekmektedir.General Tree içerisinde yapacağımız tüm işlemler için Node objelerine ihtiyaç duyarız. Bu objelerin data fieldları da “data,left,right” şeklindedir.

2 METHOD

2.1 Class Diagrams



2.2 Use Case Diagrams



2.3 Problem Solution Approach

Problemin çözümü için öncelikle yeni bir class tanımlarız. (GeneralTree<E>) Ve bu sınıfımız BinaryTree sınıfından extend edilir. Problem tanımı gereği 1. parçada oluşturulması gereken 3 metot vardır. Bunlar

levelOrderSearch: Bu metot iki parametre alır ,aldığı parametreler root ve data parametreleridir. Data parametresi aradığımız değeri temsil etmektedir. Root ise içerisinde arama yapacağımız tree ya da subtree rootunu temsil eder. Bu fonksiyon iterative ya da recursive olarak implement edilebilir. Return değeri eğer varsa data parametresinin arandığı root node olur.(Iterative olarak implement edilmiştir.)

postOrderSearch: Bu metot da levelOrderSearch gibi iki parametre alır ve bu parametreler levelOrderSearch metodundakilerle aynı işlevi görürler. Bu metot da

recursive veya iterative olarak implement edilebilir. Return değeri eğer varsa data parametresinin arandığı root node olur.(Recursive olarak implement edilmiştir.)

add: Bu metot da post veya levelOrderSearch fonksiyonları yardımı ile ekleme yapılacak root node bulunmuşsa ve eklenecek olan node GeneralTree içerisinde mevcut değilse ekleme işlemini gerçekleştirir. Parametreleri de parentItem (ekleme yapılacak anne) ve childItem(eklenecek çocuk) şeklindedir.

3 RESULT

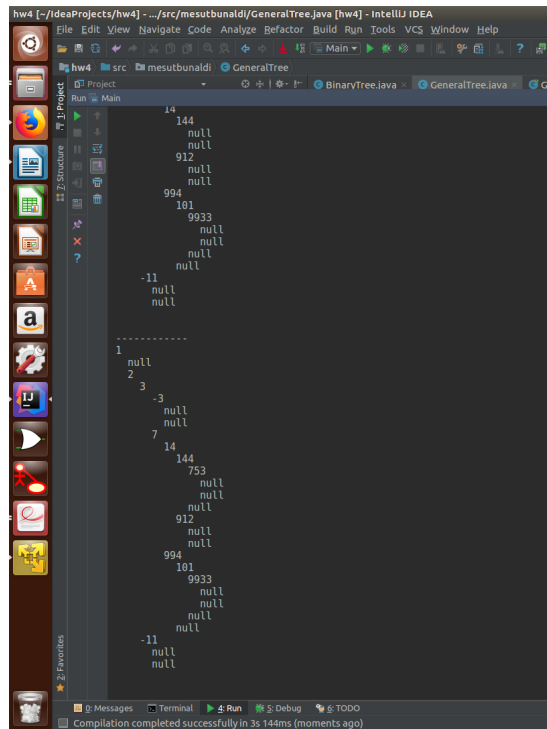
3.1 Test Cases

GeneralTree.java için test edilen fonksiyonlar :

- add()
- levelOrderSearch()
- postOrderSearch()

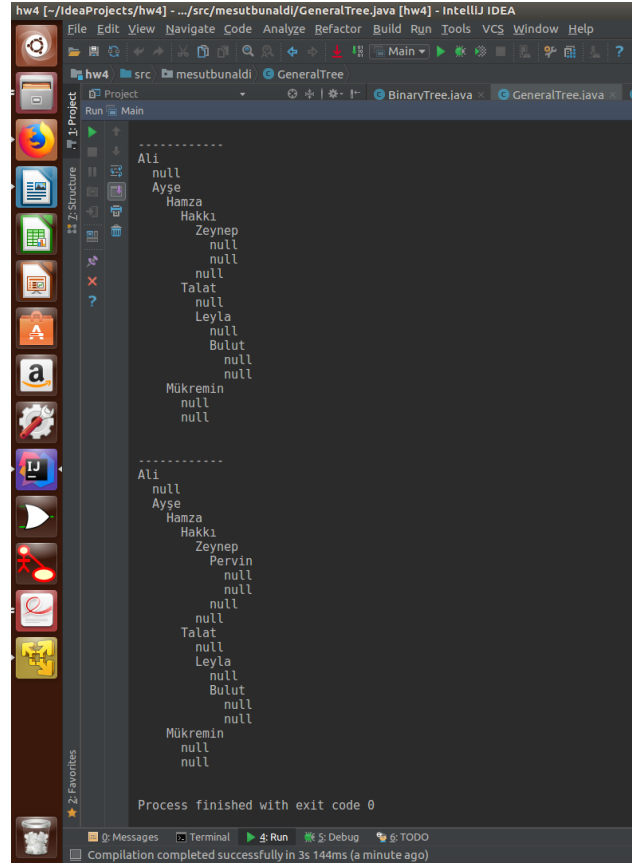
3.2 Running Results

Parça 1 için 2 farklı tipte test etme işlemi gerçekleştirilmiştir. Bunlardan birincisi test_10) fonksiyonu ile yapılmış olan Integer tipinde bir GeneralTree objesinin test işlemidir:



```
hw4 [-/ideaProjects/hw4] - .../src/mesutbunaldi/GeneralTree.java [hw4] - IntelliJ IDEA
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
hw4 src mesutbunaldi GeneralTree
Project Run Main
144
null
null
912
null
null
994
101
9933
null
null
null
-11
null
null
-----
1 null
2
3
-3
null
null
7
14
144
753
null
null
null
912
null
null
994
101
9933
null
null
null
-11
null
null
Messages Terminal Run Debug TODO
Compilation completed successfully in 36 144ms (moments ago)
```

Bunlardan ikincisi test_2() fonksiyonu ile yapılmış olan String tipinde bir GeneralTree objesinin test işlemidir:



```
-----  
Ali  
null  
Ayşe  
Hamza  
Hakkı  
Zeynep  
null  
null  
null  
Talat  
null  
Leyla  
null  
Bulut  
null  
null  
Mükrem  
null  
null  
  
-----  
Ali  
null  
Ayşe  
Hamza  
Hakkı  
Zeynep  
Pervin  
null  
null  
null  
null  
Talat  
null  
Leyla  
null  
Bulut  
null  
null  
Mükrem  
null  
null  
  
Process finished with exit code 0  
Compilation completed successfully in 3s 144ms (a minute ago)
```

Not : Ödevde istenmiş olan ikinci parça yapılamamıştır.