

Gebze Technical University  
Computer Engineering

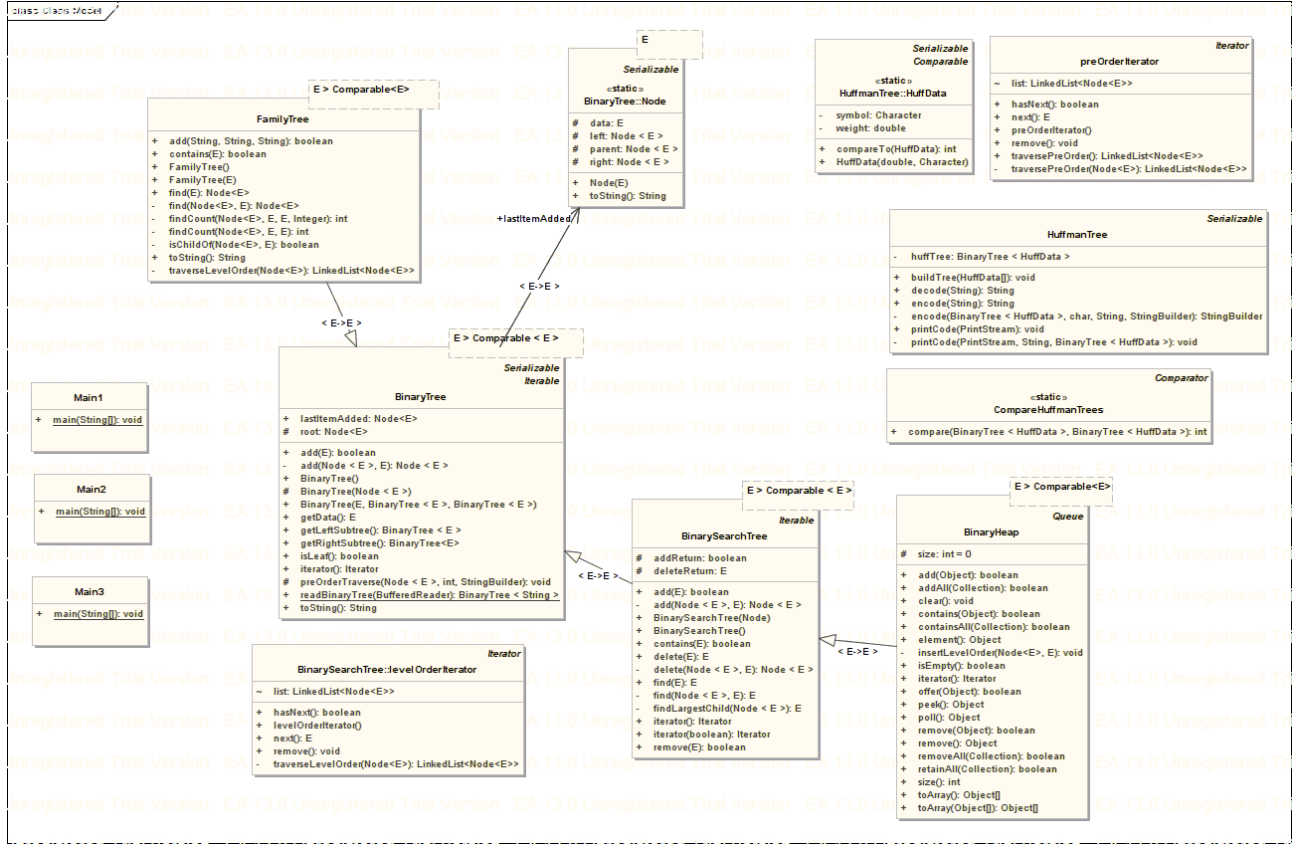
CSE 222  
2017 Spring

HOMEWORK 6 REPORT

EMRE ÇELİK  
141044024

Course Assistant:  
NUR BANU ALBAYRAK

## Class Diagrams



## Problem Solutions Approach

- 1.Part'ta BinaryHeap class'ı için BinarySearchTree extend ettim.Heap implement edebilmek için BinaryTree'de node sınıfına parent Node'u ekledim.Ayrıca BinaryTree sınıfına lastItemAdded Node'u ekledim.Bunları da kullanarak queue methodlarını implement etmeye çalıştım.Offer ve add methodlarında önce itemi ağacın eksik yerine koyup complete tree olacak şekilde ekledim ve daha sonra bu elemanı parentıyla kıyaslayıp gerekli durumda swap ede ede tepeye kadar ilerlettim.Önceliği küçük elemene verdim,yani root'ta en küçük eleman olmuş oldu.Remove veya poll yaparken de tepedeki elemene son eklenen elemanı koydum ve bunu left tree , right tree ile kıyaslaya kıyaslaya swap ederek devam ettim.
- 2.Part'ta yazdığım encode methoduna parametre olarak string verdim ve bu method içerisinde string , karakter karakter döngü içinde yardımcı bir methoda gönderiliyor ve hepsinin ayrı ayrı binary kodlanmış sonucu bir stringbuilderda toplanıyor.Sonuc olarak bu stringbuilder'ın toString() methodunu return ediyorum..
- 3.Part'ta traverse level order methodunu FamilyTree sınıfı içerisinde yazdım.Bu traverse methodu tree'nin traverse edilmiş halini linkedlist'e atıp bu listi döndürüyor.Bu methodu toString() de çağırıp toString() ' in return edeceğ stringi oluşturdum.Methodu test etmek için de main'de tree'yi build edip tree'nin toString() methodunu çağırdım.Elemanlar aralarında boşluk olacak şekilde ekrana basılmış oldu.

## Test Cases

1-) Senaryo : Huffman tree için dosyada harf ve frekans yerine , sırasıyla frekans ve harf girilmesi.  
Beklenen Durum : Programın girdiyi kabul etmemesi , sonlanması.  
Gerçekleşen Olay : Program sırasıyla string ve integer okuduğu için bu dosyada exception fırlattı.

2-) Senaryo : BinaryHeap objesi oluşturup eleman eklemekten eleman silinmeye çalışılması  
Beklenen Durum : Programın hata mesajı verip kapanması  
Gerçekleşen Olay : Program excepiton fırlattı veya null dondurdu.

3-) Senaryo : Huffman tree encode methoduna içinde boşluk barındıran kelime girilmesi.  
Beklenen Durum : Programın boşluğu kodlayamayıp kapanması.  
Gerçekleşen Olay : Program exception fırlatıp hata mesajı verdi.

The screenshot shows the IntelliJ IDEA 2016.3.5 interface. The main editor displays the code for `Main1.java`. The code defines a `BinaryHeap` class and a `Main1` class. The `Main1` class has a `main` method that creates a `BinaryHeap` object and adds several integers to it. The output window shows the execution results, which are the integers 0, 1, 2, 60, 12, 7, 11, 6, and null, indicating that the program successfully added the elements to the heap.

```
1 // BinaryHeap<Integer> heap = new BinaryHeap<>();
2
3 heap.offer(2);
4 heap.offer(3);
5 heap.offer(1);
6
7 BinaryTree.Node<Integer> n01 = new BinaryTree.Node<>{ data: 1;
8 BinaryTree.Node<Integer> n02 = new BinaryTree.Node<>{ data: 2;
9 BinaryTree.Node<Integer> n03 = new BinaryTree.Node<>{ data: 3;
10
11 n01.left = n02;
12 n01.right = n03;
13
14 BinaryHeap<Integer> heap = new BinaryHeap<>();
15 heap.offer( 7);
16 heap.offer( 60);
17 heap.offer( 12);
18 heap.offer( 11);
19 heap.offer( 0);
20 heap.offer( 6);
21 heap.offer( 4);
22 heap.offer( 12);
23 heap.offer( 2);
24
25 /*
26 System.out.println(heap + "\n");
27
28 System.out.println("poll : " + heap.poll());
29 System.out.println(heap);
30 */
31
```

Run Main1

```
0
1
2
60
null
12
null
7
null
null
11
null
6
null
null
```

