

Gebze Technical University
Computer Engineering

CSE 222
2017 Spring

HOMEWORK 8 REPORT

Seyit Ahmet KARACA
141044084

Course Assistant:Şeyma YÜCER

- Q3 -

1. Problem Solutions Approach

Kitabın source kodlarında olan add methodunu düzelttim. Delete methodu için kitabın kodu gibi bir yardımcı method kullandım. Bu methodu recursive yaptım. Bu recursive method içerisinde 3 bölüme ayrılıyor. Aynı binarySearchTree’de yaptığımız gibi istenilen eleman local root’un verisine eşitmi, büyük mü veya küçük mü karşılaştırdım. Silinecek eleman bulunduğu anda balance yapmak için önce sol ağacına bakıyor null ise sağ ağacı return ediyor, sol ağaç null ise sağ ağacı return ediyor. İki durumda değil ise solun sağını null mı diyerek kontrol ediyor, null değil ise sağın solunu kontrol ediyor. Balance değil ise balance yapıyor. Ağaçtaki veri, silinecek veriden büyük ise sol ağacı çağırıyor decrease değerine göre balance işlemlerini yapıyor. Ağaçtaki veri silinecek veriden küçük ise sağ ağacı recursive olarak tekrar çağırıyor ve decrease değerine göre balance işlemlerini yapıyor.

2. Test Cases

```
public static void main(String[] args) {
    AVLTree<String> avl = new AVLTree<>();
    avl.add("Nush");
    avl.add("ile");
    avl.add("uslanmayani");
    avl.add("etmeli");
    avl.add("tekdir");
    avl.add("tekdir");
    avl.add("ile");
    avl.add("uslanmayanin");
    avl.add("hakki");
    avl.add("kotektir");
    avl.add("edille");
    avl.add("dakik");
    avl.add("ferc");

    //System.out.println(avl.toString());

    avl.remove( target: "Nush");
    avl.remove( target: "ile");
    avl.remove( target: "uslanmayani");
    System.out.println(avl.toString());
}
```

Ödevde soru 1 de istenilen cümleyi AVLTree’ye ekledim ve silinmesi gerekenleri test olarak sildirdim.

3. Running and Results

```
0: ile
  0: etmeli
    0: dakik
      0: Nush
        null
        null
      0: edille
        null
        null
    -1: hakki
      0: ferc
        null
        null
      null
    -1: uslanmayani
      -1: tekdir
        0: kotektir
          null
          null
        null
      0: uslanmayanin
        null
        null
```

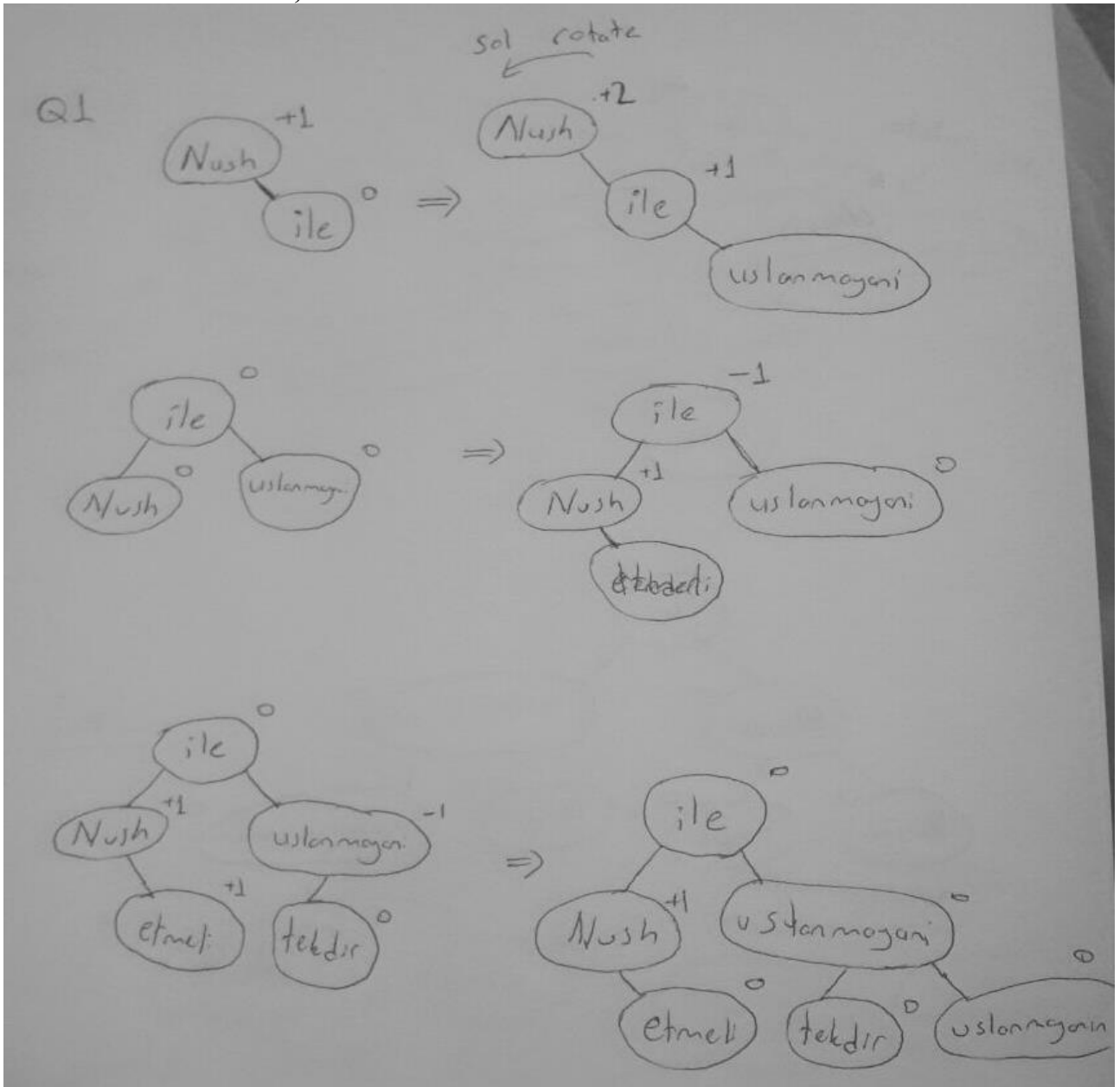
Soru 1 de istenilen AVLTree 'nin çıktısı. Hem programda hemde kağıttaki ile aynı çıktı.

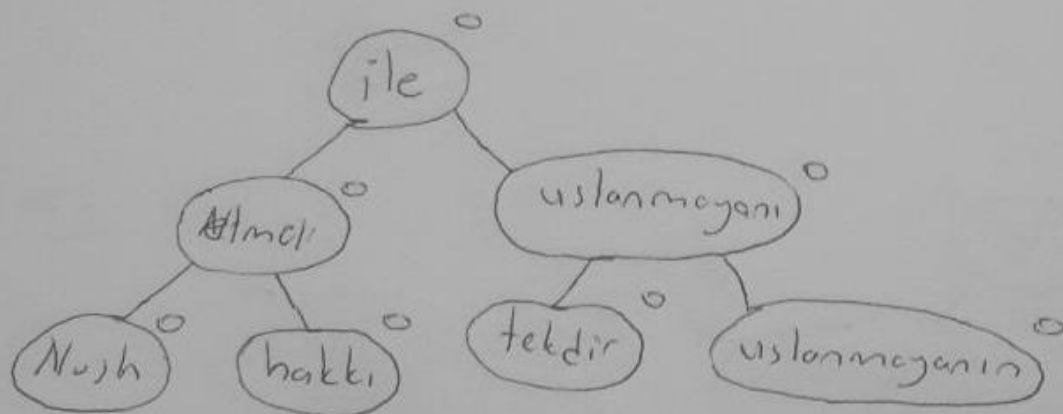
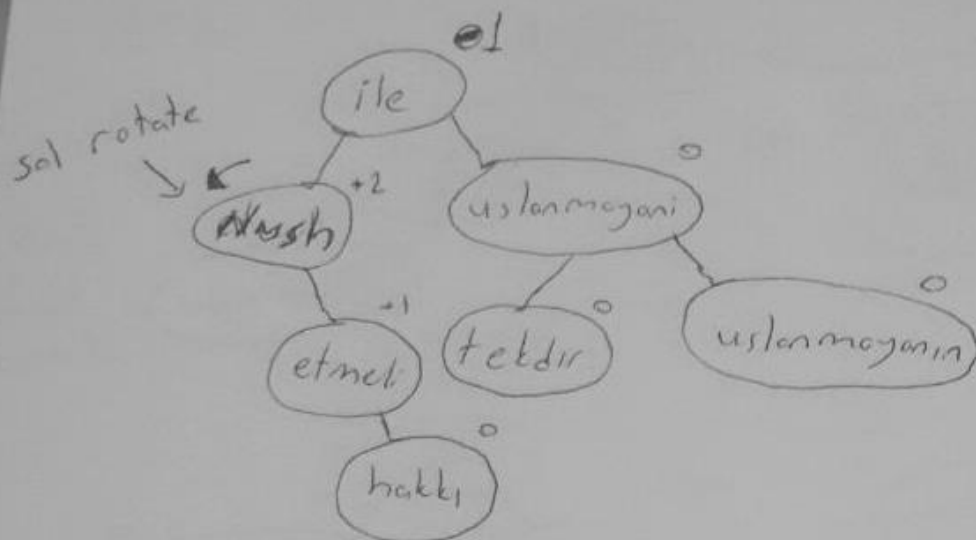
```
-1: hakki
  -1: etmeli
    1: dakik
      null
      0: edille
        null
        null
    0: ferc
      null
      null
  0: tekdir
    0: kotektir
      null
      null
    0: uslanmayanin
      null
      null
```

İlk 3 kelime silinince oluşan durum çıktıda gösterdim.

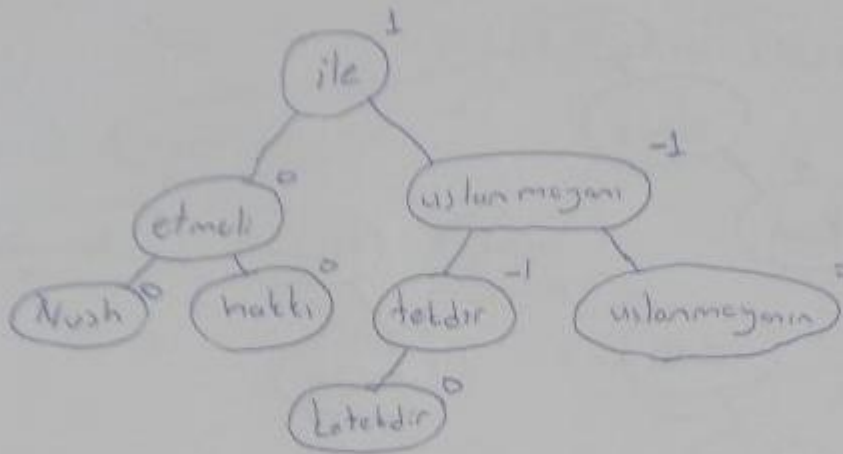
- Q1 -

Cümlenin eklenmesi ;



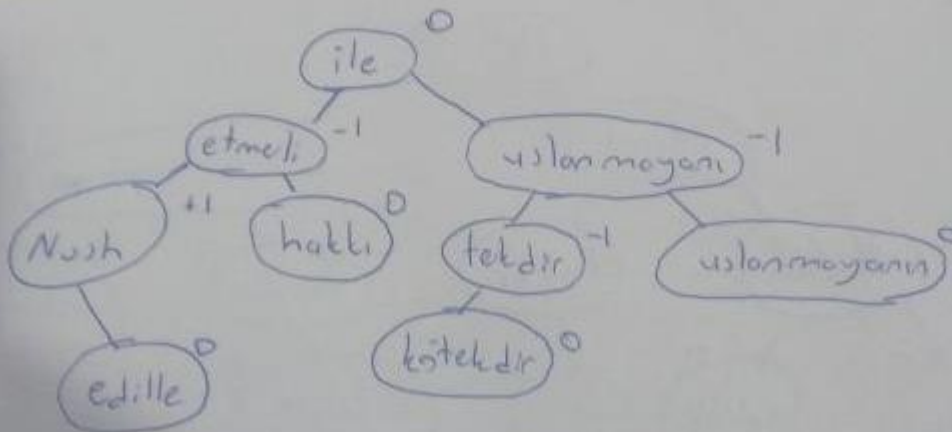


edille – dakik – ferc eklenmesi

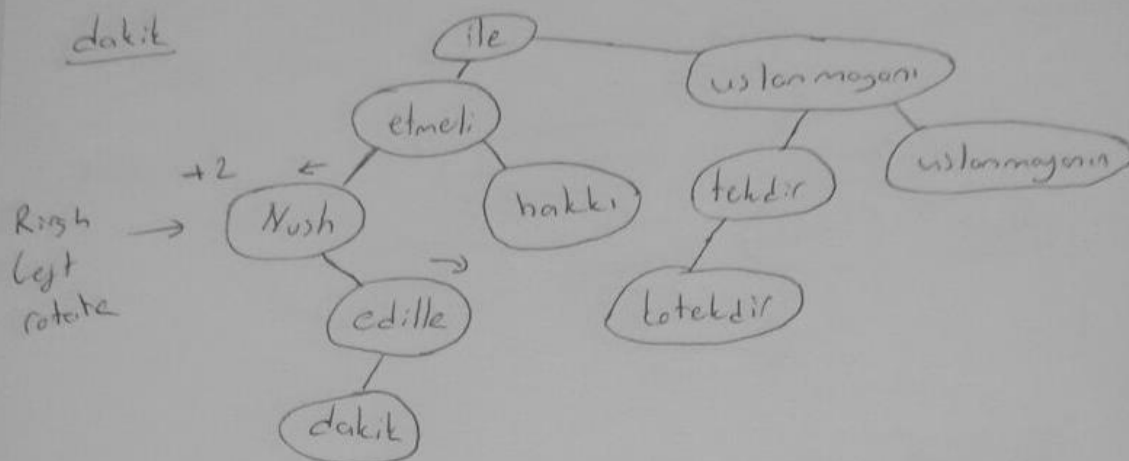


"Nush ile uslanmaganı etmeli tetdir,
tetdir ile uslanmaganın hakki kötetdir" eklendi.

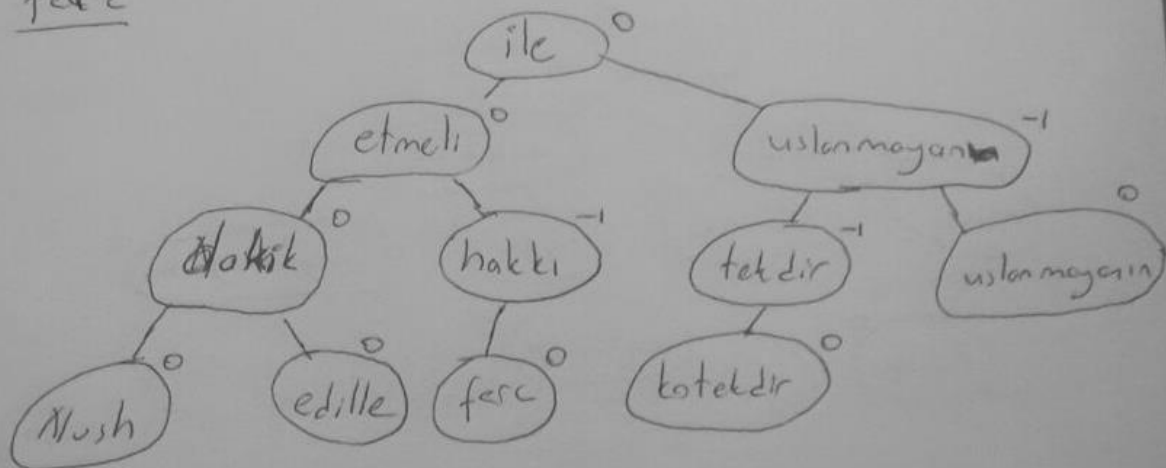
edille



dakik



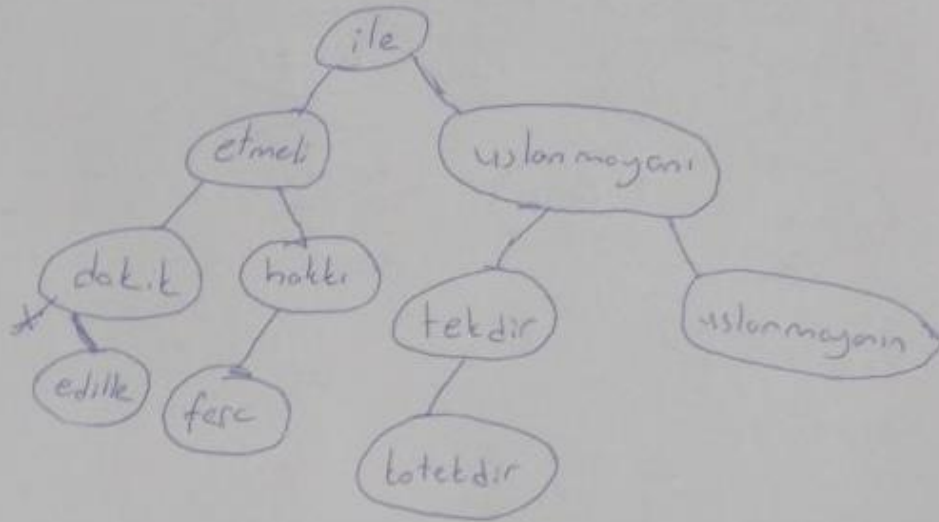
ferc



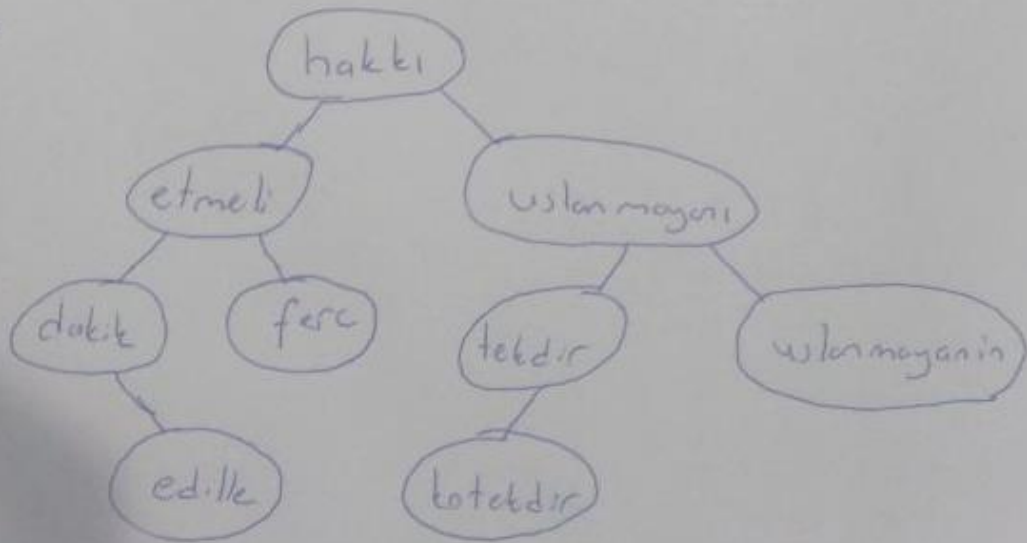
İlk 3 kelimenin silinmesi

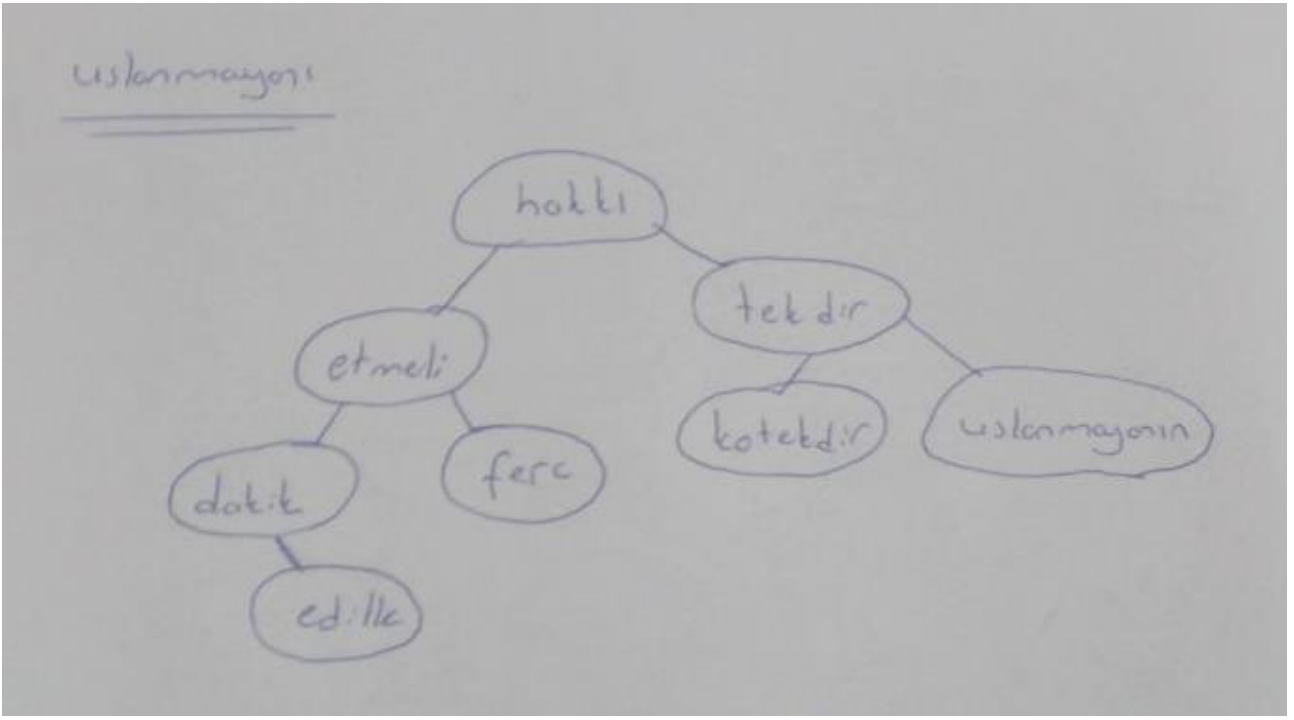
Silme işlemi; ilk üç kelime (Nush, ile, uslanmaganı)

Nush

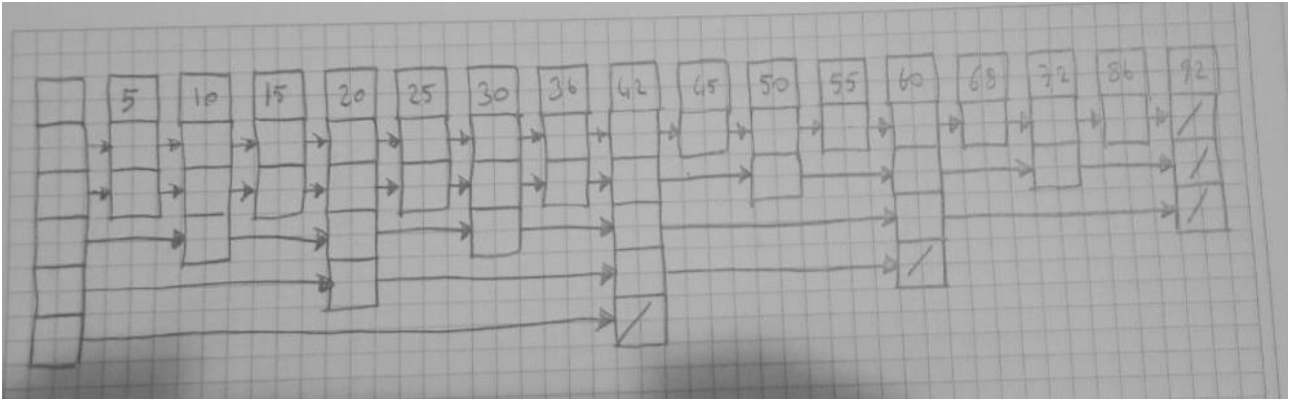


ile





--- Q2 ---



En soldaki liste head listesidir.42'nin sol tarafındaki seviyeleri 1er tane arttırdım. Yeni eleman eklendiğinde %50 ihtimal 1.seviyeye ekleneceğinden şuan için listede eleman 42 den küçükler için bir seviye önce bulunacak fakat yeni elemanlar eklenince 1.seviyede dolabilecek.Böylece aramalarda her seferinde listenin içerisindeki elemanların yarısı elenerek aramada yapılabiliyor.

Github : https://github.com/SeyitAhmetKARACA/141044084_HW8/