

## 21 Eylül 2022 -Arama Algoritmaları Ödevi

### 1.ÖDEV

İkili Arama Ağaçlarında bir değerin nasıl silineceğinin sözde kodunu, her satırda ne yapıldığını açıklayarak yazınız.

#### 1.Ödev

1-düğümToBeDeleted olsun

2-nodeToBeDeleted öğesinin rengini originalColor'da kaydedin

3- nodeToBeDeleted öğesinin sol çocuğu NULL ise

a.nodeToBeDeleted öğesinin sağ alt öğesini x öğesine atayın

b.Transplant düğümüToBeDeleted x ile

4- Else, nodeToBeDeleted öğesinin sağ çocuğu NULL ise

nodeToBeDeleted öğesinin sol çocuğunu x'e atayın.

Nakil nodeToBeDeleted, x ile.

5- else

noteToBeDeleted öğesinin minimum sağ alt ağacını y'ye atayın.

y'nin rengini orijinal Renk'e kaydedin.

y'nin sağ Çocuğunu x'e atayın.

y, nodeToBeDeleted öğesinin çocuğuyrsa, x öğesinin üst öğesini y olarak ayarlayın.

Aksi takdirde, y'yi sağ y çocuğu ile nakledin.

Nakil nodeToBeDeleted ile y.

y'nin rengini orijinal Renk ile ayarlayın.

6- Orijinal Renk SİYAH ise DeleteFix(x) çağırın.

## 2.ÖDEV

Kırmızı-Siyah Arama Ağaçlarında bir değerin nasıl silineceğinin sözde kodunu, her satırda ne yapıldığını açıklayarak yazınız.

### 2. Ödev

```
Node deleteNode(Node root, int valueToDelete) {  
    if root = null  
        return node  
    if root.value < valueToDelete  
        deleteNode(root.right, valueToDelete)  
    if root.value > valueToDelete  
        deleteNode(root.left, valueToDelete)  
    else  
        if (isLeafNode(root))  
            return null  
  
        if (root.right == null)  
            return root.left  
        if (root.left == null)  
            return root.right  
  
    else  
        minValue = findMinInRightSubtree(root)  
        root.value = minValue  
        removeDuplicateNode(root)
```

return root

Sözde kod:

düğüm boşsa ->

düğümü döndür (ya bize boş bir ağaç verildi ya da

ağacı tekrarladık ve değeri bulamadık,

bu durumda orijinal ağacı döndürürüz)

mevcut düğüm aradığımız değerden küçükse ->

sağ tarafta özyinelemeli çağrı işlevi

geçerli düğüm değerden büyükse ->

sol tarafta özyinelemeli çağrı işlevi

aksi halde silmek istediğimiz düğümü bulduk

-> düğümün çocuğu yoksa null döndür

-> düğümün bir çocuğu varsa, boş olmayan alt ağacı döndür

-> aksi halde:

-> sağ alt ağaçta minimum değeri bulun

-> mevcut düğümün değerini bulunan minimum değere ayarlayın

-> şimdi çoğaltılan düğümü sil

-> mevcut düğümü döndür