

"İkili Ağaçtan Düğüm Silme Algoritmasında İyileştirmeler" konulu Veri Yapıları uygulama dersinde 2 iyileştirme uygulaması yapılacaktır:

1. Uygulama

Bilindiği gibi ikili ağaçtan çocuklu düğümler silinirken yerine ya kendisinden büyük en küçük (KBK) ya da kendisinden küçük en büyük (KKB) düğüm yazılır. Bu uygulamada ağaçtan silinecek düğümün sol alt ağacındaki ve sağ alt ağacındaki toplam düğüm sayıları karşılaştırılır. Sol alt ağaçtaki düğüm sayısı fazla ise KKB; sağ alt ağaçtaki düğüm sayısı fazla ise de KBK uygulanır. Alt ağaçlardaki düğüm sayıları, ağaçta gezinme fonksiyonlarından herhangi birinde küçük bir değişiklik yapılarak hesaplanabilir.

2. Uygulama

Bu uygulamada düğüm (node) structure'ına sol ve sağ alt ağaçlardaki düğümlerin sayısını tutacak birer int değişken eklenir ve düğümler ağaca eklenirken değişkenler ++ yapılarak güncellenir. Herhangi bir düğüm silinirken bu değişkenler karşılaştırılır ve KKB veya KBK yaklaşımlarından hangisinin uygulanacağına karar verilir.

Uygulamalar için şablon projeye [buradan](#) ulaşabilirsiniz.

Örnek çıktılarına da [buradan](#) ulaşabilirsiniz.

```
C:\Users\Omer_Cakir\Desktop\1. Uygulama\Debug\07 Binary Tree.exe
Number of left subNodes = 3 and right subNodes = 2 before deleting = 4 so replace with KKB
Number of left subNodes = 5 and right subNodes = 4 before deleting = 8 so replace with KKB
Number of left subNodes = 1 and right subNodes = 2 before deleting = 12 so replace with KBK
Number of left subNodes = 8 and right subNodes = 11 before deleting = 16 so replace with KBK
Number of left subNodes = 1 and right subNodes = 2 before deleting = 20 so replace with KBK
Number of left subNodes = 3 and right subNodes = 5 before deleting = 24 so replace with KBK
Number of left subNodes = 1 and right subNodes = 2 before deleting = 28 so replace with KBK
```