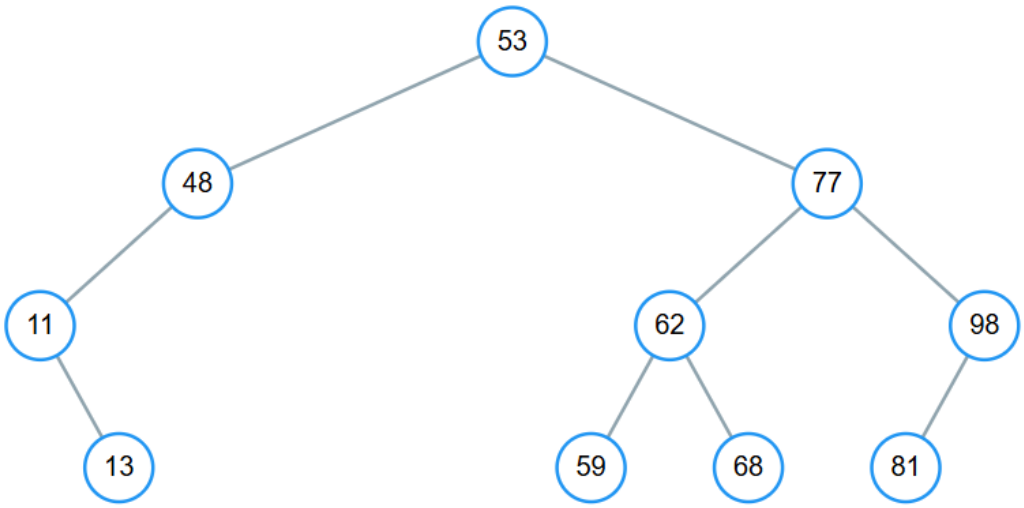
GİRESUN ÜNİVERSİTESİ MÜHENDİSLİK FAKÜLTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ
VERİ YAPILARI DERSİ FİNAL SINAVI

Adı – Soyadı – Numarası:

Soru 1: Aşağıda verilen ikili ağaç için (a) kök önde (preorder), (b) kök ortada (inorder) ve (c) kök sonda (postorder) gezildiğinde oluşan çıktıyı yazınız. 53 numaralı düğüm kök düğümdür. (d) Kök ortada gezmenin çıktısı bize neyi verir?



Inorder traversal: 11 -> 13 -> 48 -> 53 -> 59 -> 62 -> 68 -> 77 -> 81 -> 98

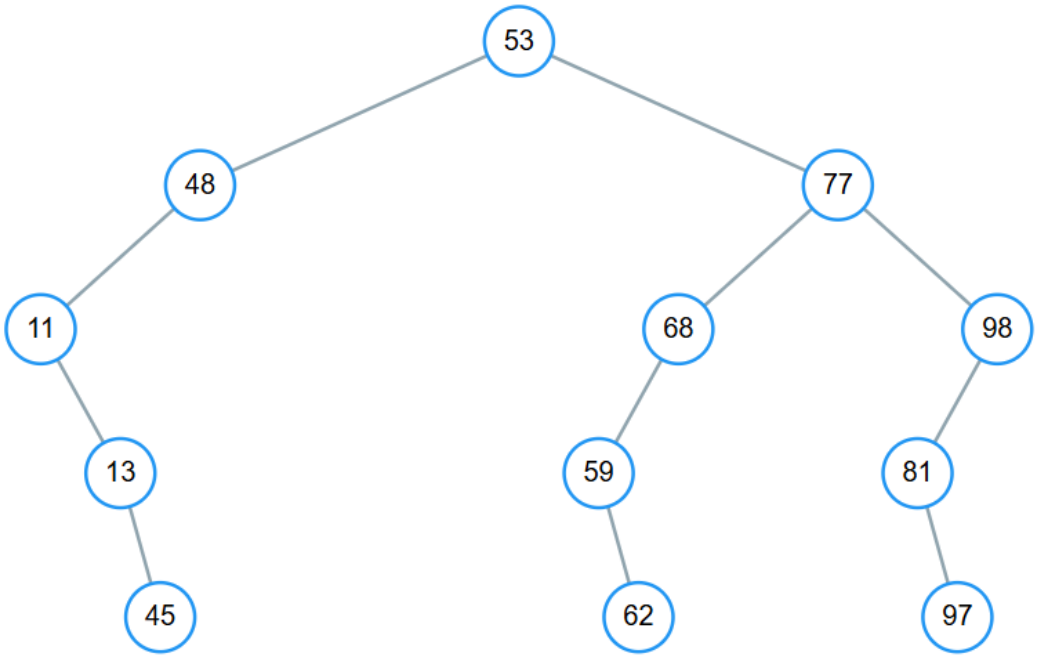
Preorder traversal: 53 -> 48 -> 11 -> 13 -> 77 -> 62 -> 59 -> 68 -> 98 -> 81

Postorder traversal: 13 -> 11 -> 48 -> 59 -> 68 -> 62 -> 81 -> 98 -> 77 -> 53

Kök ortada gezinme (in-order traversal), özellikle ikili arama ağaçlarında (BST) uygulandığında, düğümleri artan sırada (sorted order) verir. Bu, sol alt ağaç → kök → sağ alt ağaç sırasıyla gerçekleşir ve ağacın sıralı bir listesini üretir.

Soru 2: Yukarıda verilen ikili arama ağacına, belirtilen işlemler sırayla uygulandıktan sonraki halini çiziniz. (a) ekle 97, ekle 45, sil 62, ekle 62. (b) Silme işlemi sırasında ağacın yapısı hangi durumlarda bozulabilir? Ne tür önlem alınabilir?

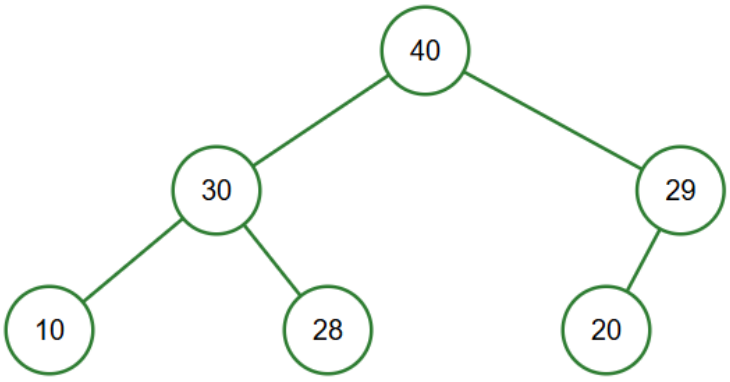
Başarılar.. Dr. Öğr. Üyesi Sercan KÜLCÜ 06.01.2026



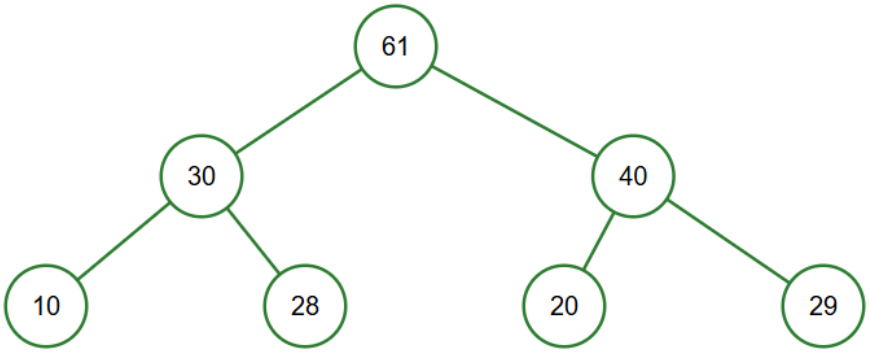
Silme işlemi, özellikle ikili arama ağaçları (BST), AVL ağaçları veya kırmızı-siyah ağaçlar gibi veri yapılarında, ağacın yapısını (dengesini, sıralamasını veya bütünlüğünü) etkileyebilir.

Silinen düğümün iki çocuğu varsa ve yerine geçen düğüm (genellikle sağ alt ağacın en küçük düğümü) doğru yerleştirilmezse, BST özelliği (sol çocuk < kök < sağ çocuk) bozulabilir.

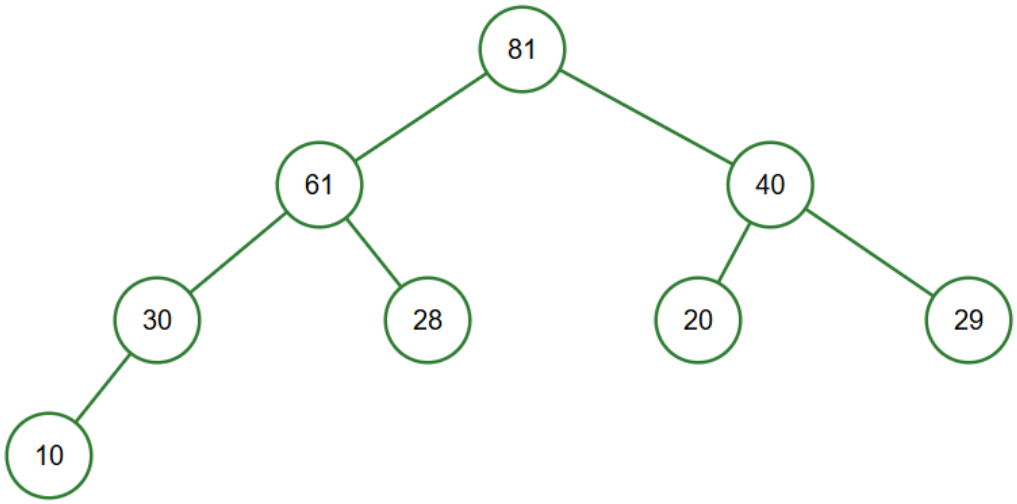
Soru 3: Aşağıda verilen maksimum ikili yığına (max-heap) verilen işlemler uygulandıktan sonraki halini çiziniz. (a) ekle 61, ekle 81, sil (extract max), sil (extract max), ekle 61. (b) Maksimum yığını hangi işlem veya işlemlerde kullanmak avantajlıdır.



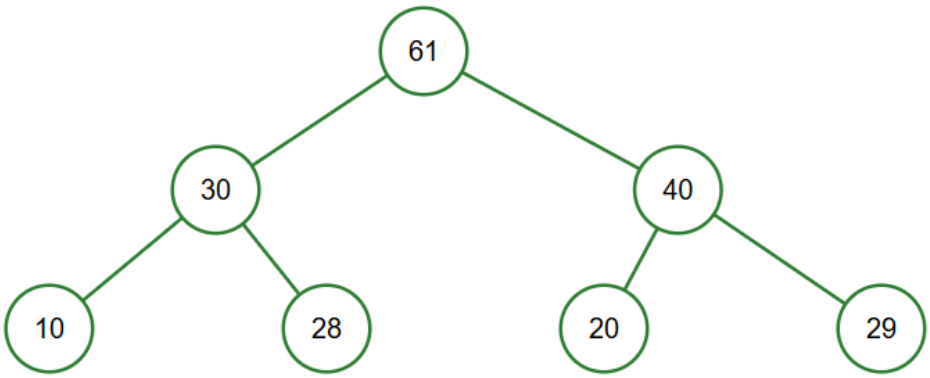
61 eklendikten sonra



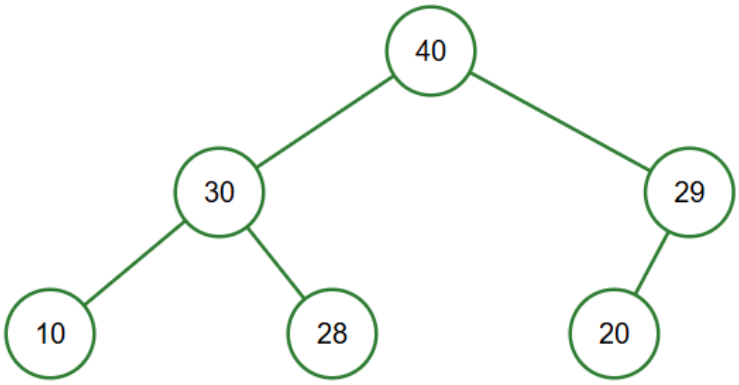
81 eklendikten sonra



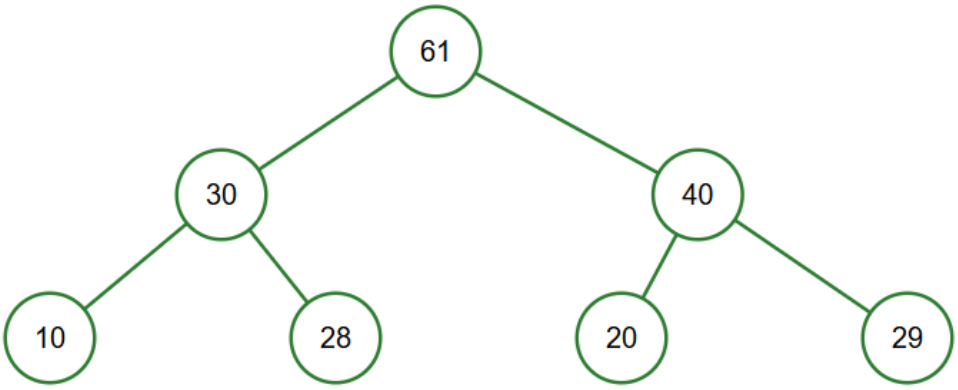
Extract max sonrası



Extract max sonrası



61 eklendikten sonrası

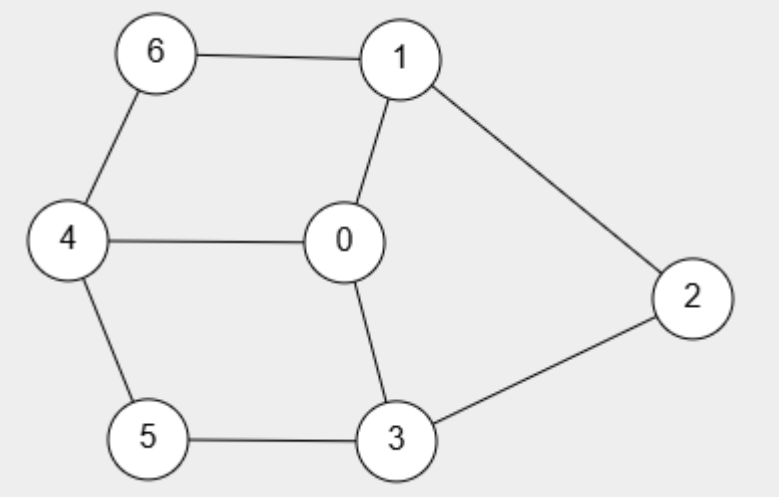


Maksimum yığın (max-heap), bir tam ikili ağaç yapısıdır ve kök düğüm her zaman en büyük elemanı içerir; her ebeveyn düğüm, çocuklarından büyük veya eşit olmalıdır. Bu yapı, belirli işlemlerde avantaj sağlar çünkü ekleme (insert), en büyük elemanı çıkarma (extract-max) ve yığın oluşturma (heapify) gibi işlemler O(log n) veya O(n) zaman karmaşıklığında gerçekleştirilebilir.

Örnek Kullanımlar: İşletim sistemlerinde süreç önceliklendirme (highest priority first). Oyunlarda skor tablosu yönetimi (en yüksek skoru hızlıca bulma). Büyük veri setlerini sıralamak, özellikle bellek kısıtlı ortamlarda. Tavsiye sistemlerinde en popüler K ürünü bulma, istatistiksel analizlerde en yüksek değerleri seçme.

Soru 4: Aşağıda verilen yönsüz çizgeyi (a) komşuluk matrisi ve komşuluk listesi ile gösteriniz. (b) Derinlik Öncelikli (DFS) ve (c) Genişlik Öncelikli (BFS) arama ile gezildiğinde oluşan yolu (path) yazınız. Komşuların seçilme sırası küçükten büyüğe doğru olacaktır. Aramaya 0 nolu düğümden başlanacaktır. (d) DFS ve BFS hangi durumda aynı çıktıyı üretir?

GİRESUN ÜNİVERSİTESİ MÜHENDİSLİK FAKÜLTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ
VERİ YAPILARI DERSİ FİNAL SINAVI



Adj. Matrix

0	1	0	1	1	0	0
1	0	1	0	0	0	1
0	1	0	1	0	0	0
1	0	1	0	0	1	0
1	0	0	0	0	1	1
0	0	0	1	1	0	0
0	1	0	0	1	0	0

Adj. List

0 -> 1, 4, 3
1 -> 0, 2, 6
2 -> 1, 3
3 -> 2, 5, 0
4 -> 5, 0, 6
5 -> 3, 4
6 -> 4, 1

BFS: 0 → 1 → 3 → 4 → 2 → 6 → 5

DFS: 0 → 1 → 2 → 3 → 5 → 4 → 6

Hiç düğüm yoksa veya sadece bir düğüm varsa, her iki algoritma da aynı boş veya tek elemanlı çıktıyı verir (örneğin, [] veya [A]).

Çizge düğümlerin sırayla bağlı olduğu lineer bir yapıdaysa (örneğin, A → B → C → D, dallanma veya döngü olmadan), her iki algoritma da aynı sırayı takip eder.

Başarılar.. Dr. Öğr. Üyesi Sercan KÜLCÜ 06.01.2026