

**T.C.  
BANDIRMA ONYEDİ EYLÜL ÜNİVERSİTESİ  
MÜHENDİSLİK VE DOĞA BİLİMLERİ FAKÜLTESİ**



**VERİ YAPILARI PROJE ÖDEVİ  
Dr. Öğr. Üyesi Fahrettin Burak DEMİR**

**INSERTION SORT  
Ve  
BFS (Genişlik Öncelikli Arama)**

**2211505010 Uğur ELMA**

**Yazılım Mühendisliği Bölümü**

**Güz Dönemi 2024-2025**

## İçindekiler

<b>1. INSERTION SORT .....</b>	<b>3</b>
1.1 Nedir? .....	3
1.2 Yöntemi.....	3
1.3 Kağıt Üzerinde Örnek ve Çözümü .....	3
1.4 Kod Üzerinde Örnek ve Çözümü, Çıktısı .....	5
<b>2. BFS (Genişlik Öncelikli Arama) .....</b>	<b>6</b>
2.1 Nedir? .....	6
2.2 Yöntemi.....	6
2.3 Kağıt Üzerinde Örnek ve Çözümü .....	7
2.4 Kod Üzerinde Örnek ve Çözümü, Çıktısı .....	8
<b>3. KODLARIN BAĞLANTISI .....</b>	<b>8</b>

## 1. INSERTION SORT

### 1.1 Nedir?

Genelde Quick ve Merge Sort' a göre daha az miktarda belirli bir veri grubunu soldan sağa doğru büyükten küçüğe sıralamakta kullanılan en fazla  $n$  (veri grubu sayısı) adımda çözen Sıralama Algoritmalarından birisidir.

### 1.2 Yöntemi

İlk olarak veri grubundan soldan başlayarak ikili grup alırız burada küçük olanı sola büyük olanı sağa koyarız (her zaman grubun en sağındaki veri diğerleri ile karşılaştırılır) ve 1. adımı tamamlarız. Sonra yeni oluşan sıralamada soldan başlayarak üçlü grup alırız bu grup içinde soldan sağa küçükten büyüğe sıralarız ve 2. adımı tamamlarız. Sonra dördü grup için yaparız ve 3. adımı tamamlarız ve bu olaylar özyinelemeli (recursive) olarak birbirini en fazla  $n$ ' e kadar tekrar eder yani veri grubu soldan sağa küçükten büyüğe doğru sıralanana kadar devam eder, gerekirse bir adım daha yapıp en soldaki verinin diğer bütün verilerden küçük olduğu kanıtlanır, ve böylece bize verilen veri grubunu soldan sağa doğru büyükten küçüğe sıralamış oluruz.

### 1.3 Kağıt Üzerinde Örnek ve Çözümü

Verilen dizi: 3, 8, 15, 4, 23, 2, 39, 5, 12, 10

Beklenen cevap: 2, 3, 4, 5, 8, 10, 12, 15, 23, 39

Bulunan cevap: 2, 3, 4, 5, 8, 10, 12, 15, 23, 39

Sayfa-1:

Handwritten notes showing the steps of Insertion Sort on a list of numbers:

- Initial list: 3, 8, 15, 4, 23, 2, 39, 5, 12, 10
- Step 1: 3 < 8, so 3 stays in place. (1st step complete)
- Step 2: 8 < 15, so 8 stays in place. (2nd step complete)
- Step 3: 15 > 4, so they swap. (3rd step complete)
- Step 4: 8 > 4, so they swap. (4th step complete)
- Step 5: 3 < 4, so 3 stays in place. (5th step complete)
- Step 6: 15 < 23, so 15 stays in place. (6th step complete)

The final sorted list is: 2, 3, 4, 5, 8, 10, 12, 15, 23, 39.

Sayfa-2:

3 4 8 15 23 → 2 39 5 12 10  
 ⇒ 23 > 2 olduğundan yer değiştirirler.  
 3 4 8 15 → 2 23 39 5 12 10  
 ⇒ 15 > 2 olduğundan yer değiştirirler.  
 3 4 8 → 2 15 23 39 5 12 10  
 ⇒ 8 > 2 olduğundan yer değiştirirler.  
 3 4 → 2 8 15 23 39 5 12 10  
 ⇒ 4 > 2 olduğundan yer değiştirirler.  
 3 → 2 4 8 15 23 39 5 12 10  
 ⇒ 3 > 2 olduğundan yer değiştirirler ve 5. adım biter.  
 1 3 4 8 15 23 → 39 5 12 10  
 ⇒ 23 < 39 olduğundan aynı kalır ve 6. adım tamamlanır.  
 1 2 3 4 8 15 23 39 → 5 12 10  
 ⇒ 39 > 5 olduğundan yer değiştirirler.  
 1 2 3 4 8 15 23 → 5 39 12 10  
 ⇒ 23 > 5 olduğundan yer değiştirirler.  
 1 2 3 4 8 15 → 5 23 39 12 10  
 ⇒ 15 > 5 olduğundan yer değiştirirler.  
 1 2 3 4 8 → 5 15 23 39 12 10  
 ⇒ 8 > 5 olduğundan yer değiştirirler.  
 1 2 3 4 → 5 8 15 23 39 12 10  
 ⇒ 4 < 5 olduğundan aynı kalır ve 7. adım tamamlanır.

Sayfa-3:

1 3 4 5 8 15 23 39 → 12 10  
 ⇒ 39 > 12 olduğundan yer değiştirirler.  
 1 2 3 4 5 8 15 23 → 12 39 10  
 ⇒ 23 > 12 olduğundan yer değiştirirler.  
 1 2 3 4 5 8 15 → 12 23 39 10  
 ⇒ 15 > 12 olduğundan yer değiştirirler.  
 1 2 3 4 5 8 → 12 15 23 39 10  
 ⇒ 8 < 12 olduğundan aynı kalır ve 8. adım tamamlanır.  
 1 2 3 4 5 8 12 15 23 39 → 10  
 ⇒ 39 > 10 olduğundan yer değiştirirler.

Sayfa-4:

2 3 4 5 8 12 15 23 >10 39  
⇒ 23 > 10 olduğundan yer değiştirirler.

2 3 4 5 8 12 15 >10 23 39  
⇒ 15 > 10 olduğundan yer değiştirirler.

2 3 4 5 8 12 >10 15 23 39  
⇒ 12 > 10 olduğundan yer değiştirirler.

2 3 4 5 8 >10 12 15 23 39  
⇒ 8 < 10 olduğundan aynen kalır ve 9. adım tamamlanır.

2 3 4 5 8 10 12 15 23 39

#### 1.4 Kod Üzerinde Örnek ve Çözümü, Çıktısı

Çıktı-1:

Verilen dizi: 3, 8, 15, 4, 23, 2, 39, 5, 12, 10

Beklenen cevap: 2, 3, 4, 5, 8, 10, 12, 15, 23, 39

3 8 15 4 23 2 39 5 12 10  
⇒ 3 < 8 olduğundan aynen kalır ve 1. adım tamamlanır.

3 8 15 4 23 2 39 5 12 10  
⇒ 8 < 15 olduğundan aynen kalır ve 2. adım tamamlanır.

3 8 15 4 23 2 39 5 12 10  
⇒ 15 > 4 olduğundan yer değiştirirler.

3 8 4 15 23 2 39 5 12 10  
⇒ 8 > 4 olduğundan yer değiştirirler.

3 4 8 15 23 2 39 5 12 10  
⇒ 3 < 4 olduğundan aynen kalır ve 3. adım tamamlanır.

3 4 8 15 23 2 39 5 12 10  
⇒ 15 < 23 olduğundan aynen kalır ve 4. adım tamamlanır.

3 4 8 15 23 2 39 5 12 10  
⇒ 23 > 2 olduğundan yer değiştirirler.

3 4 8 15 2 23 39 5 12 10  
⇒ 15 > 2 olduğundan yer değiştirirler.

3 4 8 2 15 23 39 5 12 10  
⇒ 8 > 2 olduğundan yer değiştirirler.

3 4 2 8 15 23 39 5 12 10  
⇒ 4 > 2 olduğundan yer değiştirirler.

3 2 4 8 15 23 39 5 12 10  
⇒ 3 > 2 olduğundan yer değiştirirler ve 5. adım tamamlanır.

2 3 4 8 15 23 39 5 12 10  
⇒ 23 < 39 olduğundan aynen kalır ve 6. adım tamamlanır.

2 3 4 8 15 23 39 5 12 10  
⇒ 39 > 5 olduğundan yer değiştirirler.



Çıktı-2:

```
2 3 4 8 15 23 5 39 12 10
=> 23 > 5 olduğundan yer değiştiriler.
2 3 4 8 15 5 23 39 12 10
=> 15 > 5 olduğundan yer değiştiriler.
2 3 4 8 5 15 23 39 12 10
=> 8 > 5 olduğundan yer değiştiriler.
2 3 4 5 8 15 23 39 12 10
=> 4 < 5 olduğundan aynen kalır ve 7. adım tamamlanır.
2 3 4 5 8 15 23 39 12 10
=> 39 > 12 olduğundan yer değiştiriler.
2 3 4 5 8 15 23 12 39 10
=> 23 > 12 olduğundan yer değiştiriler.
2 3 4 5 8 15 12 23 39 10
=> 15 > 12 olduğundan yer değiştiriler.
2 3 4 5 8 12 15 23 39 10
=> 8 < 12 olduğundan aynen kalır ve 8. adım tamamlanır.
2 3 4 5 8 12 15 23 39 10
=> 39 > 10 olduğundan yer değiştiriler.
2 3 4 5 8 12 15 23 10 39
=> 23 > 10 olduğundan yer değiştiriler.
2 3 4 5 8 12 15 10 23 39
=> 15 > 10 olduğundan yer değiştiriler.
2 3 4 5 8 12 10 15 23 39
=> 12 > 10 olduğundan yer değiştiriler.
2 3 4 5 8 10 12 15 23 39
=> 8 < 10 olduğundan aynen kalır ve 9. adım tamamlanır.
```

Bulunan Cevap: 2 3 4 5 8 10 12 15 23 39  
Ana sayfaya dönmek için klavyeden herhangi bir tuşa basınız!

## 2. BFS (Genişlik Öncelikli Arama)

### 2.1 Nedir?

Birbiriyle bağlantılı belirli bir gruptaki tüm elemanları genişlik öncelikli arama, aynı düzeydeki verilerin hepsine sırasıyla uğrayıp sonra alt düzeye geçme, ile dolaşmaya yarayan Graf Algoritmalarından birisidir.

### 2.2 Yöntemi

İlk olarak veri grubundaki başlangıç veriyi belirliyoruz, bu veri genelde ortada ya da solda ebeveyni olmayan olur, genelde bu verinin bulunduğu düzeyde sadece kendisi olduğundan bu veri gezilen veriler listesine eklenir ve bütün verilerin gezilip gezilmediği kontrol edilir eğer hepsi gezildiyse arama biter eğer hepsi gezilmediyse bir alt düzeye geçilir, genelde soldakinden başlanır, o düzeydeki veriler sırasıyla gezilir ve gezilen veriler listeye eklenir sonra bu

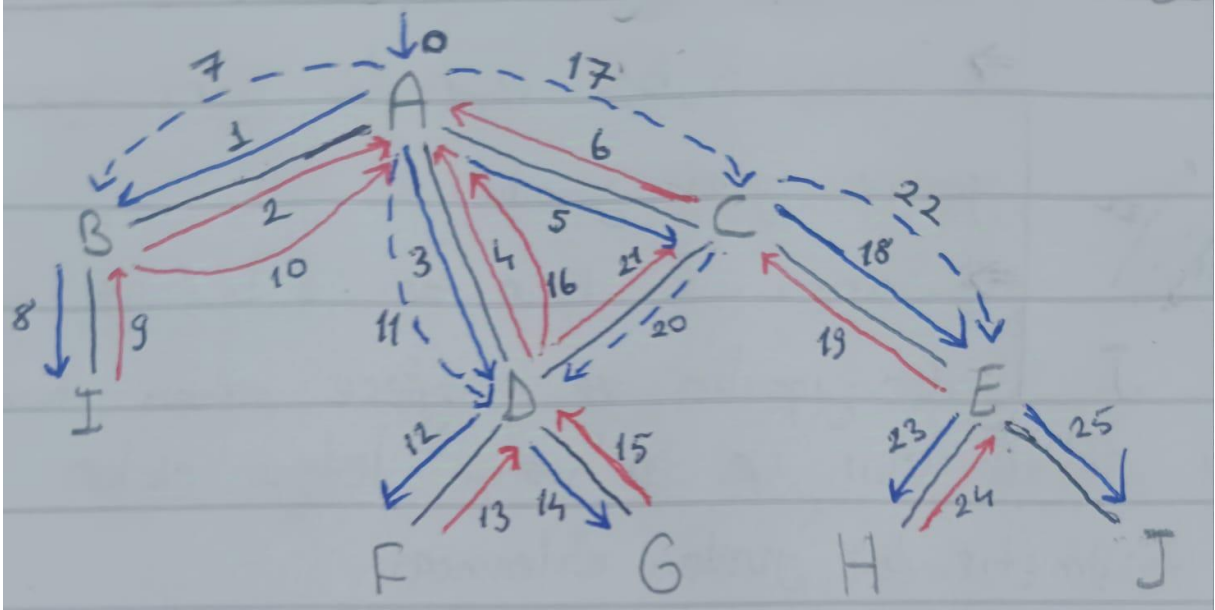
düzeydeki verilerin hepsine aynı işlem önceki veriye yapıldığı gibi ayrı ayrı özyinemeli (recursive) olarak gerçekleştirilir. Bütün veriler gezildiğinde ise tarama işlemi biter.

### 2.3 Kağıt Üzerinde Örnek ve Çözümü

Verilen dizi: A, B, C, D, E, F, G, H, I, J

Beklenen sıra: A, B, D, C, I, F, G, E, H, J

Sayfa-1:



Sayfa-2:

- 0 → A getirilir ve listeye eklenir. {A}
- 1 → Bir alt düzeydeki elemanlardan B getirilir ve listeye eklenir. {A, B}
- 2 → Aynı düzeydeki gezilmeyenler için ebeveyne (A) geri dönülür.
- 3 → Bir alt düzeydeki elemanlardan D getirilir ve listeye eklenir. {A, B, D}
- 4 → Aynı düzeydeki gezilmeyenler için ebeveyne (A) geri dönülür.
- 5 → Bir alt düzeydeki elemanlardan C getirilir ve listeye eklenir. {A, B, D, C}
- 6 → Aynı düzeydeki gezilmeyenler için ebeveyne (A) geri dönülür.
- 7 → Bir alt düzeydeki veri (B) gezildiği için listeye eklenmez ama o veriye gidilir.
- 8 → Bir alt düzeydeki eleman olan I getirilir ve listeye eklenir. {A, B, D, C, I}
- 9 → Aynı düzeydeki gezilmeyenler için ebeveyne (B) geri dönülür.
- 10 → Bir alt düzeyde gezilecek veri kalmadığından ebeveyne (A) dönülür.
- 11 → Bir alt düzeydeki veri (D) gezildiği için listeye eklenmez ama o veriye gidilir.
- 12 → Bir alt düzeydeki eleman olan F getirilir ve listeye eklenir. {A, B, D, C, I, F}
- 13 → Aynı düzeydeki gezilmeyenler için ebeveyne (D) geri dönülür.
- 14 → Bir alt düzeydeki eleman olan G getirilir ve listeye eklenir. {A, B, D, C, I, F, G}
- 15 → Aynı düzeydeki gezilmeyenler için ebeveyne (D) geri dönülür.

Sayfa-3:

16 → Bir alt düzeyde gezilecek veri kalmadığından ebeveyne (A) dönülür.  
17 → Bir alt düzeydeki veri (C) gezildiği için listeye eklenmez ama o veriye gidilir.  
18 → Bir alt düzeydeki eleman olan E gezilir ve listeye eklenir. {A,B,D,C,I,F,G,E}  
19 → Aynı düzeydeki gezilmeyenler için ebeveyne (C) geri dönülür.  
20 → Bir alt düzeydeki veri (D) gezildiği için listeye eklenmez ama o veriye gidilir.  
21 → Aynı düzeydeki gezilmeyenler için ebeveyne (C) geri dönülür.  
22 → Bir alt düzeydeki veri (E) gezildiği için listeye eklenmez ama o veriye gidilir.  
23 → Bir alt düzeydeki eleman olan H gezilir ve listeye eklenir. {A,B,D,C,I,F,G,E,H}  
24 → Aynı düzeydeki gezilmeyenler için ebeveyne (E) geri dönülür.  
25 → Bir alt düzeydeki eleman olan J gezilir ve listeye eklenir. {A,B,D,C,I,F,G,E,H,J}  
Böylece bütün veriler listeye eklenmiş oldu. Bulunan Sıra: A,B,D,C,I,F,G,E,H,J

## 2.4 Kod Üzerinde Örnek ve Çözümü, Çıktısı

Çıktı:

Verilen dizi: A, B, C, D, E, F, G, H, I, J

Beklenen sıra: A, B, D, C, I, F, G, E, H, J

A verisi gezildi ve listeye eklendi.  
B verisi gezildi ve listeye eklendi.  
D verisi gezildi ve listeye eklendi.  
C verisi gezildi ve listeye eklendi.  
I verisi gezildi ve listeye eklendi.  
F verisi gezildi ve listeye eklendi.  
G verisi gezildi ve listeye eklendi.  
D verisi daha önceden gezildiği için listeye eklenmedi.  
E verisi gezildi ve listeye eklendi.  
H verisi gezildi ve listeye eklendi.  
J verisi gezildi ve listeye eklendi.

Bulunan sıra: A, B, D, C, I, F, G, E, H, J

Ana sayfaya dönmek için klavyeden herhangi bir tuşa basınız!

## 3. KODLARIN BAĞLANTISI

<https://github.com/UgurElma/VeriYapilariProje.git>