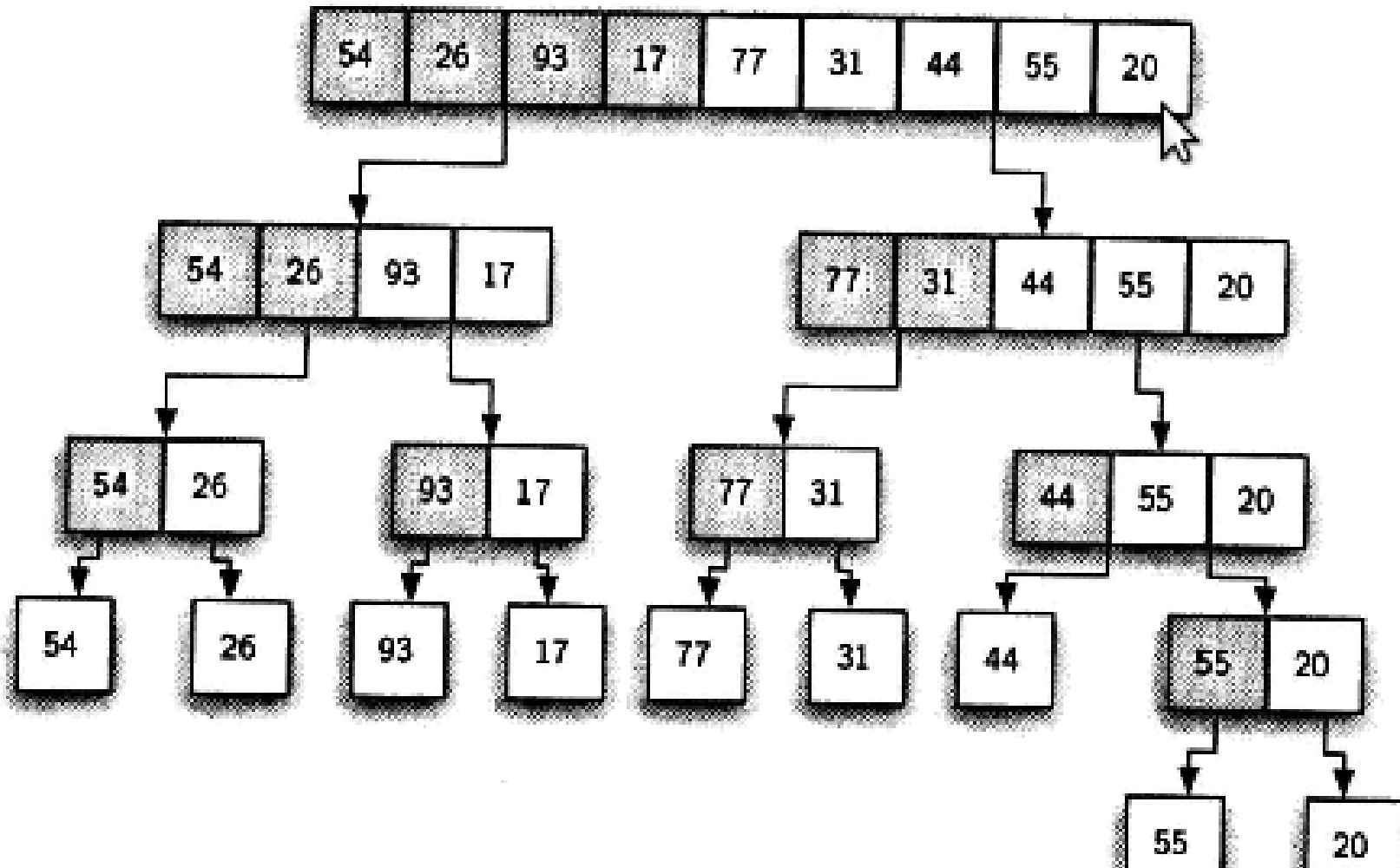


4.4.5 birleştirmeli (merge) sıralama

- böl-idare et yaklaşımı sıralama algoritmalarının başarımını arttırabilir
- merge sort özyinelidir
- sürekli olarak listeyi ikiye böl
- bir elemanlı liste baz durumudur
- yarılanmış listeyi meger sort ile sırala

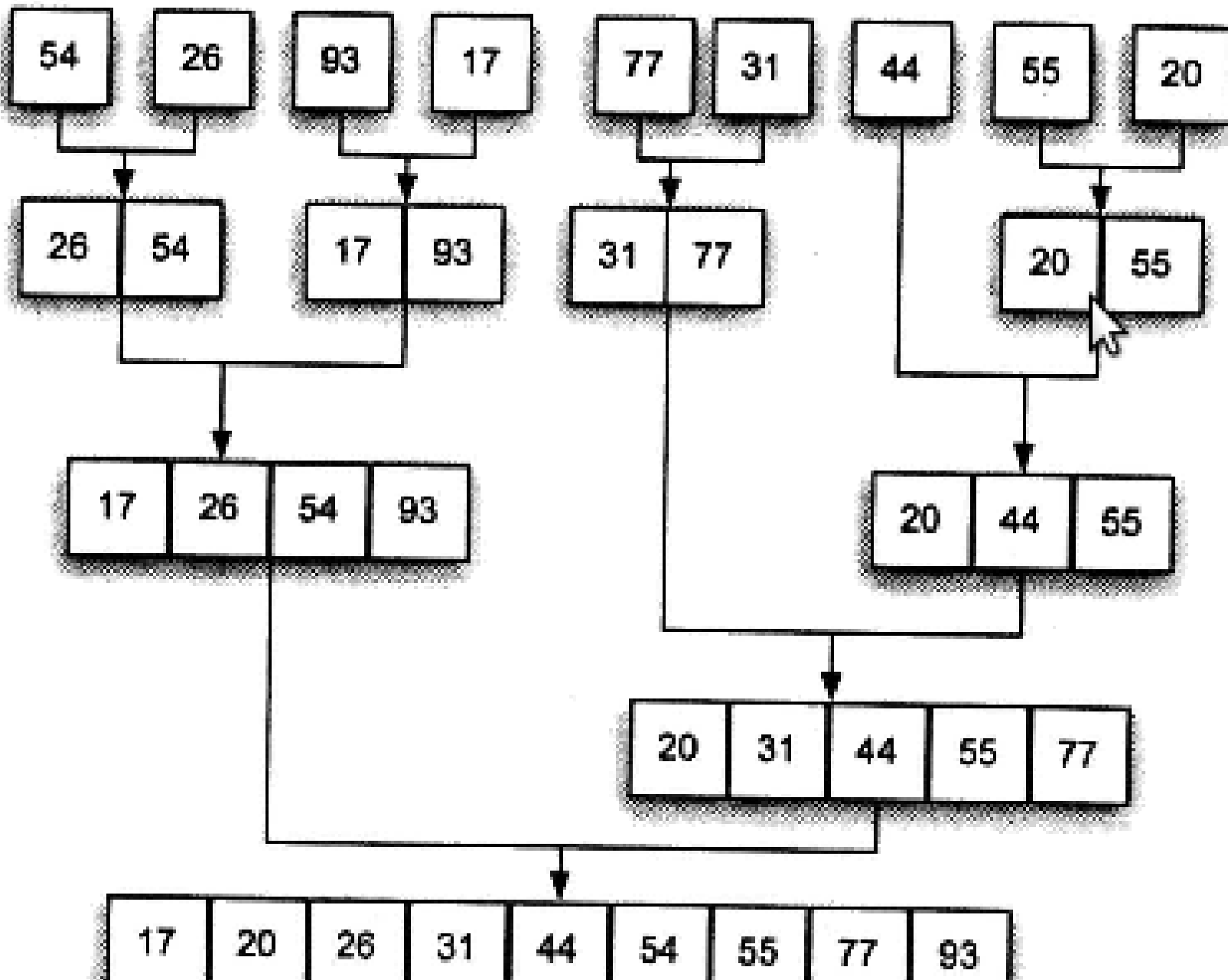
böl

→ altlisteleri sırala



birleştir

→ sıralı altlisteleri birleştir



gerçekleme

gerçekleme: split

```
1      def mergeSort(alist):
2          print "Splitting ",alist
3          if len(alist)>1:
4              mid = len(alist)/2
5              lefthalf = alist[:mid]
6              righthalf = alist[mid:]
7
8              mergeSort(lefthalf)
9              mergeSort(righthalf)
```

gerçekleme

gerçekleme: merge

```
1         i=0
2         j=0
3         k=0
4         while i<len(lefthalf) and j<len(righthalf):
5             if lefthalf[i]<righthalf[j]:
6                 alist[k]=lefthalf[i]
7                 i=i+1
8             else:
9                 alist[k]=righthalf[j]
10                j=j+1
11            k=k+1
12
13        while i<len(lefthalf):
14            alist[k]=lefthalf[i]
15            i=i+1
16            k=k+1
17
18        while j<len(righthalf):
19            alist[k]=righthalf[j]
20            j=j+1
21            k=k+1
22        print "Merging ",alist
```

açıklama

→ listenin sağ-sol parçalarını dilimlemeyle elde et

```
1 lefthalf = alist[:mid]
2 righthalf = alist[mid:]
```

→ sağ-sol altlisteleri merge sort

```
1 mergeSort(lefthalf)
2 mergeSort(righthalf)
```

birleştirme

→ i değişkeni sol altliste, j değişkeni sağ altliste için

→ nihayi liste indisi- k

→ eğer sol altlistedeki değer daha küçükse

```
1  alist[k]=lefthalf[i]
```

→ değilse

```
1  alist[k]=righthalf[j]
```

→ sol altlistede eleman kaldıysa, onları nihayi listeye koy

→ sağ altlistede eleman kaldıysa, onları nihayi listeye koy

demo

→ demo

```
1  >>> from listing_4_22 import *
2  >>> alist = [54, 26, 93, 17, 77, 31, 44, 55, 20]
3  >>> mergeSort(alist)
4  Splitting [54, 26, 93, 17, 77, 31, 44, 55, 20]
5  Splitting [54, 26, 93, 17]
6  Splitting [54, 26]
7  Splitting [54]
8  Merging [54]
9  Splitting [26]
10 Merging [26]
11 Merging [26, 54]
12 Splitting [93, 17]
13 Splitting [93]
14 Merging [93]
15 Splitting [17]
16 Merging [17]
17 Merging [17, 93]
18 Merging [17, 26, 54, 93]
```

→ demo

```
1  Merging [17, 20, 26, 31, 44, 54, 55, 77, 93]
```

→ demo

```
1  Splitting [77, 31, 44, 55, 20]
2  Splitting [77, 31]
3  Splitting [77]
4  Merging [77]
5  Splitting [31]
6  Merging [31]
7  Merging [31, 77]
8  Splitting [44, 55, 20]
9  Splitting [44]
10 Merging [44]
11 Splitting [55, 20]
12 Splitting [55]
13 Merging [55]
14 Splitting [20]
15 Merging [20]
16 Merging [20, 55]
17 Merging [20, 44, 55]
18 Merging [20, 31, 44, 55, 77]
```


analiz

- yarılama (split), ikil aramadaki gibi $O(\log n)$
- birleştir (merge), $O(n)$
- sonuçta n parçanın her biri $O(\log n)$
- shell karmaşıklığı $O(n \log n)$
- ekstradan bellek talep etmez
- eleman sayısı (n) büyüdükçe bu önem kazanır