



ALGORITMI I STRUKTURE PODATAKA

STUDIJSKI PROGRAMI:

SOFTVERSKO INŽENJERSTVO, RAČUNARSKA TEHNIKA, INFORMATIKA I MATEMATIKA

NASTAVNIK: DOC. DR ULFETA MAROVAC, UMAROVAC@NP.AC.RS

ALGORITMI I STRUKTURE PODATAKA

METODE ZAMENE

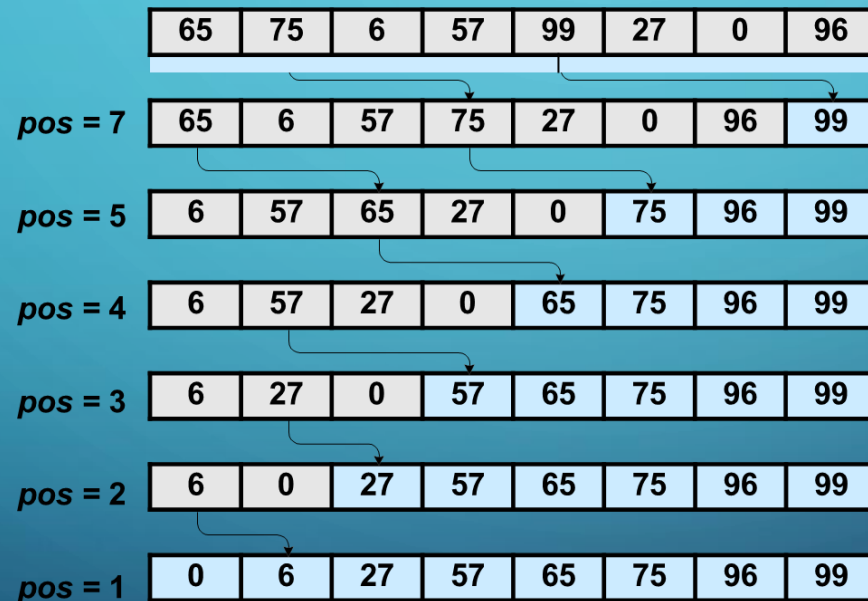
- Princip– zamena mesta dva elementa koji nisu u pravilnom poretku
- U ovoj grupi metoda zamena je glavni mehanizam na kojem se zasniva sortiranje. Zamena dva elementa se obavlja uvek kada se utvrdi da oni nisu u pravilnom poretku.
- U zavisnosti od toga da li se zamenjuju susedni elementi ili udaljeni elementi i na koji način, razlikuju se:
 - **metode direktne zamene,**
 - **particijskog sortiranja i**
 - **pobitnog razdvajanja.**

DIREKTNA ZAMENA (BUBBLESORT)

- Zamena mesta dva susedna elementa
- Najveći element izađe na početak uređenog dela
- Optimizacije
- najviša pozicija na kojoj je bila zamena u prolazu
- kraj – prolaz bez zamena

BUBBLESORT

- Promenljiva *pos* pamti u svakom koraku najvišu poziciju



BUBBLESORT

```
BUBBLESORT(a)  
pos = n  
repeat  
    bound = pos  
    pos = 0  
    for i = 1 to bound - 1 do if  
        (a[i] > a[i + 1]) then  
            a[i]  $\leftrightarrow$  a[i + 1]  
            pos = i  
        end_if  
    end_for  
until pos = 0
```

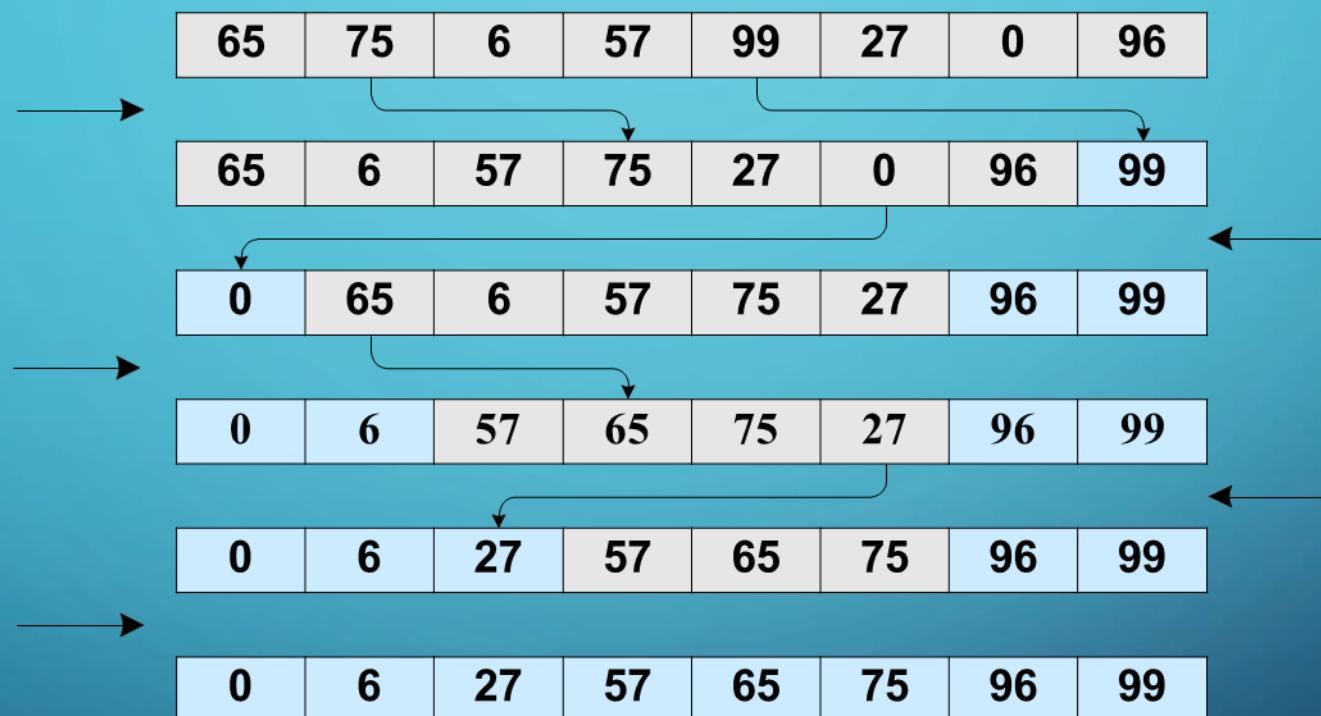
PERFOMANSE

- Najbolji slučaj – sortiran niz
- $M_{\min} = 0, C_{\min} = n - 1 \Rightarrow O(n)$
- Najgori slučaj – obrnuto sortiran niz
- $M_{\max} = C_{\max} = 0.5(n^2 - n) \Rightarrow O(n^2)$
- Prosečni slučaj
- $C_{\text{ave}} = 0.5(n^2 - n \cdot \ln(n)), M_{\text{ave}} = 0.25(n^2 - n) \Rightarrow O(n^2)$

POBOLJŠANJE

- Shakersort
- Alternativno menja smer prolaska
- Manje poređenja, ali isti red složenosti

SHAKERSORT



QUICKSORT

- Poređenja i zamene na većoj udaljenosti
- Particijsko sortiranje (Hoare)
- Razdvojni element (pivot)
- Donja particija +pivot +gornja particija
- Rekurzivna podela na particije do jedinične veličine

QUICKSORT

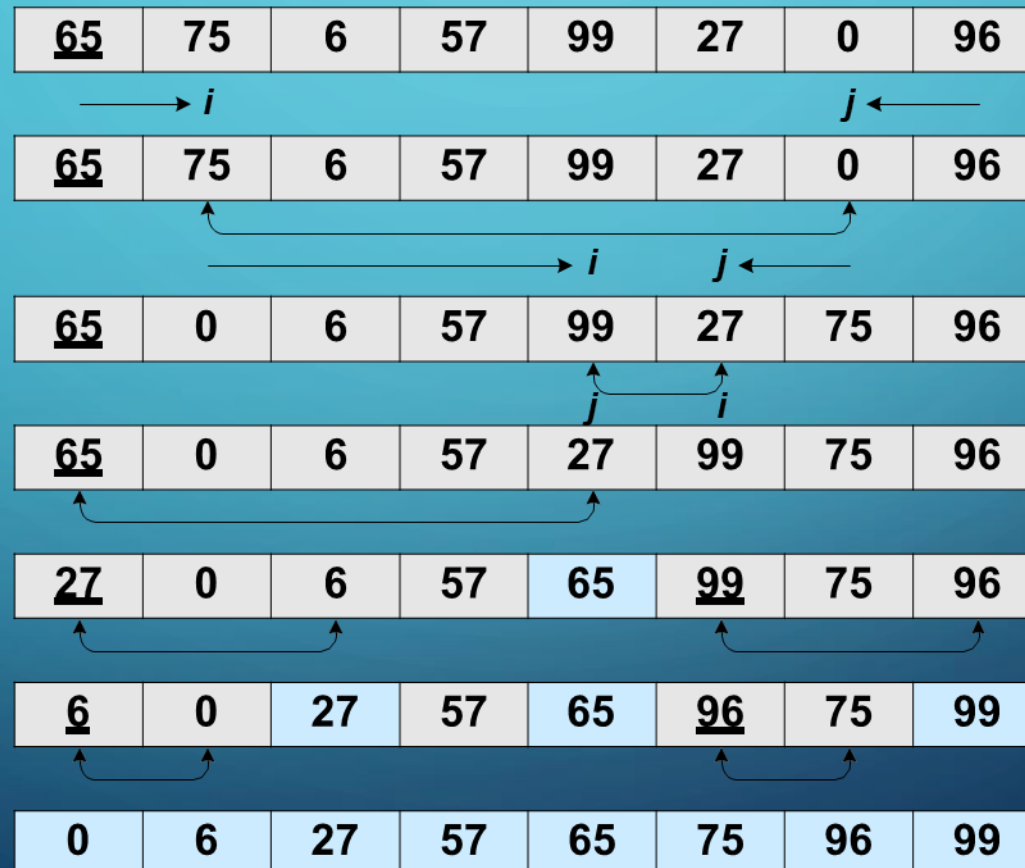
```
QUICKSORT(a, low, high) if  
(low < high) then  
  j = PARTITION(a, low, high)  
  QUICKSORT(a, low, j - 1)  
  QUICKSORT(a, j + 1, high)  
end_if
```

QUICKSORT

```
PARTITION(a, down, up)  
i = down  
j = up  
pivot = a[down] while (i < j)  
do  
    while ((a[i] ≤ pivot) and (i < j)) do  
        i = i + 1 end_while  
    while (a[j] > pivot) do  
        j = j - 1  
    end_while  
    if (i < j) then  
        a[i] ↔ a[j]  
    end_if  
end_while
```

```
a[down] = a[j]  
a[j] = pivot  
return j
```

QUICKSORT



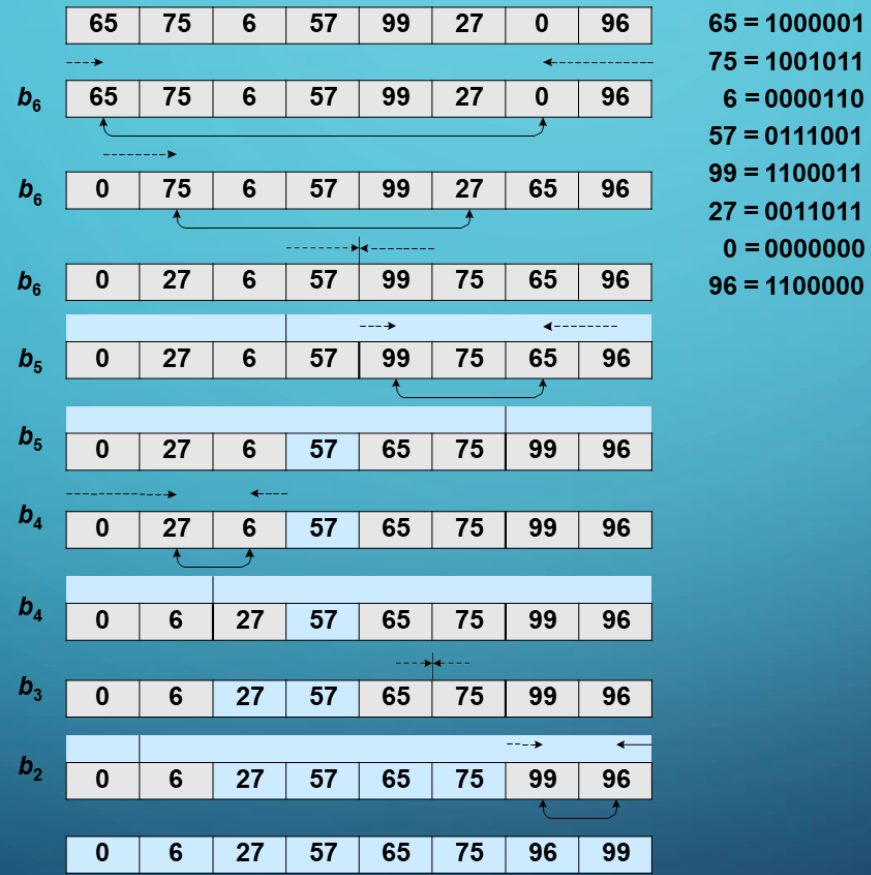
QUICKSORT

- Iterativna realizacija sa stekom
- Najbolji slučaj – jednake particije $\Rightarrow O(n \log n)$
- Najgori slučaj – jedna particija $\Rightarrow O(n^2)$
- Prosečan slučaj gori od najboljeg za 38% $\Rightarrow O(n \log n)$
- Veliki uticaj izbora pivota
- slučajan izbor
- srednji od tri (ili više) elementa particije
- srednja vrednost (meansort)

POBITNO RAZDVAJANJE

- Radix exchange
- Binarna reprezentacija ključa $(b_{m-1} b_{m-2} \dots b_0)_2$
- Počinje se od najstarijeg bita
- Donja ($b_i = 0$) i gornja ($b_i = 1$) particija
- Složenost – $O(n \log n)$

POBITNO RAZDVAJANJE



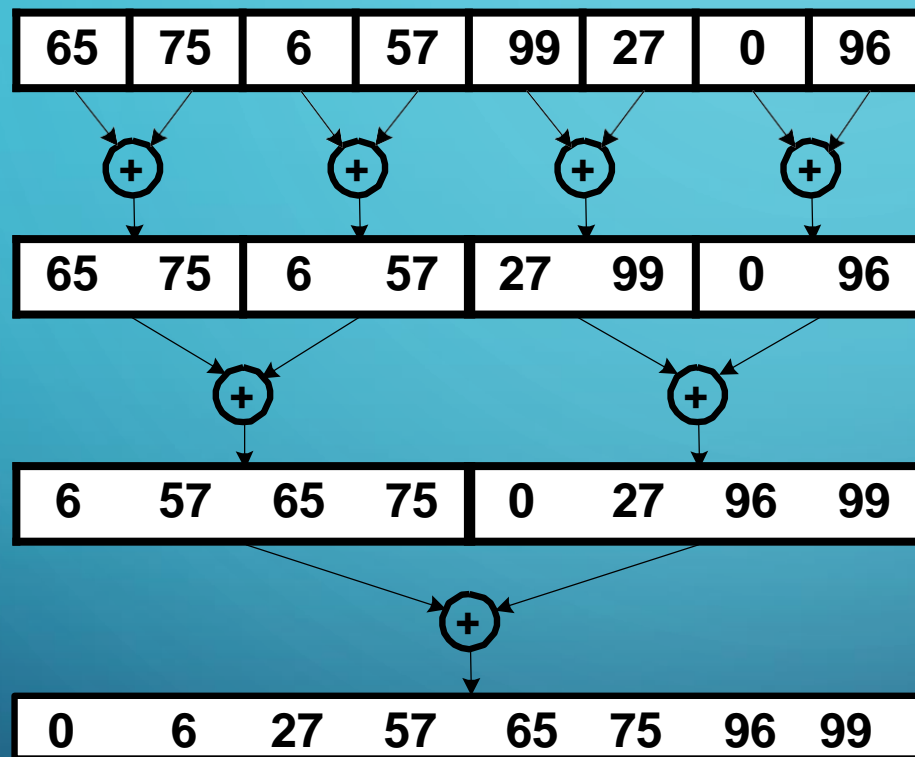
METODI SPAJANJA

- Ove metode su pretežno karakteristicne za spoljasnje sortiranje mada se mogu koristiti i kod unutrašnjeg sortiranja.

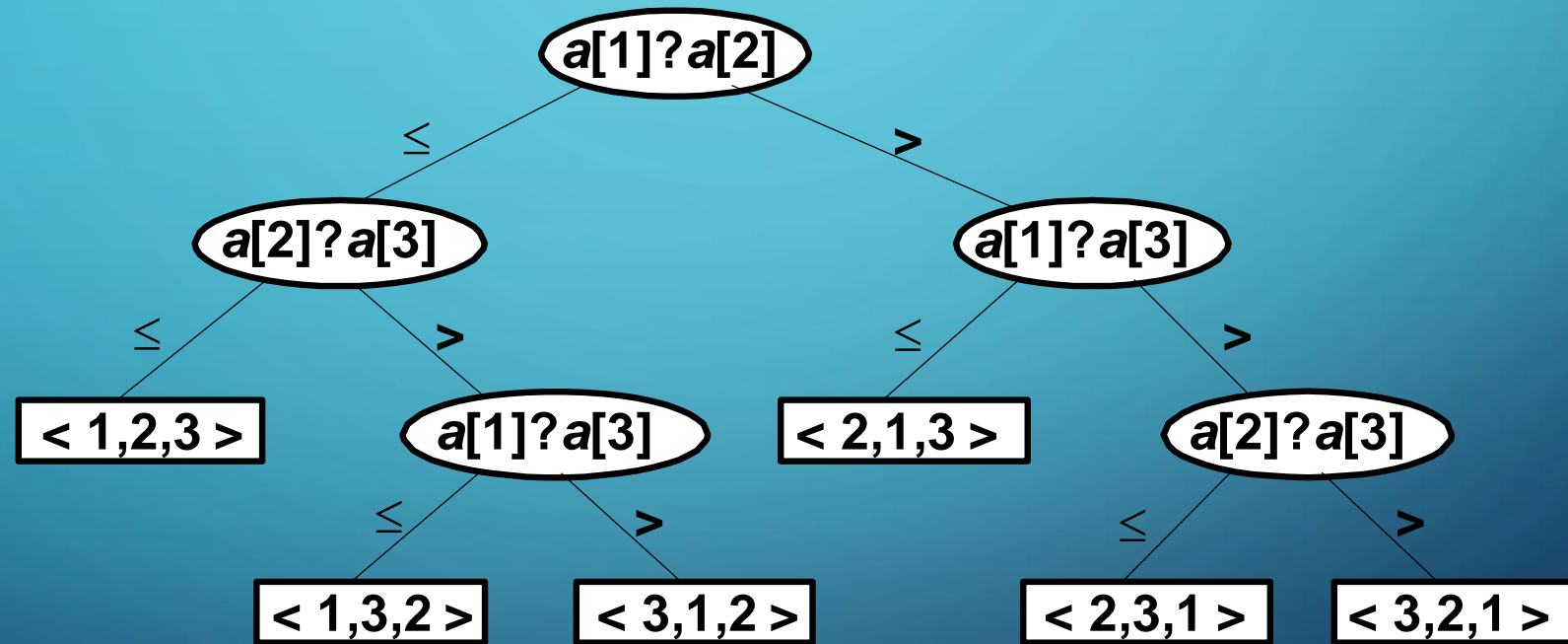
DIREKTNO SPAJANJE

- Susedni elementi se spoje u uređene dvojke, pa u uređene četvorke, ...
- Potreban pomoćni niz
- Implementacija sa ulančanim listama
- izbegava premeštanje
- Performanse
 - ✓ broj prolaza $\lceil \log n \rceil \Rightarrow O(n \log n)$
 - ✓ garantovane performanse

DIREKTNO SPAJANJE



PERFORMANCE



PERFORMANCE

- Stablo odlučivanja
 - ✓ čvorovi predstavljaju poređenja
 - ✓ listovi predstavljaju moguće sortirane poretke
 - ✓ visina stabla predstavlja najgori slučaj
- $I = 2^n = n!$ $n! > (n/e)^n$
- $h \geq \log n! > \log(n/e)^n = n \log n - n \log e \Rightarrow O(n \log n)$
- Garantovane performanse u najgorem slučaju ne mogu biti bolje od $O(n \log n)$
- Prosečna performansa – $PE/e \Rightarrow O(n \log n)$

TEST PITANJA

1. Koji je princip metoda zamene?
2. Koji tipovo metoda zamene postoje?
3. Na prethodnom primeru primenite shakersort
4. Kako se moze optimizovati bubblesort?
5. Navesti primer sortiranja bubblesort-om
6. Koji je princip rada i performanse quicksort algoritma?
7. Navesti primer sortiranja pomocu ovog algoritma.
8. Navesti primer sortiranja pobitnim razdvajanjem
9. Koji je princip sotiranja kod metoda spajanja?
10. Sta je direktno spajanje kako se izvodi?