

# Strukture podataka i algoritmi

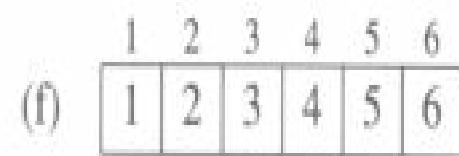
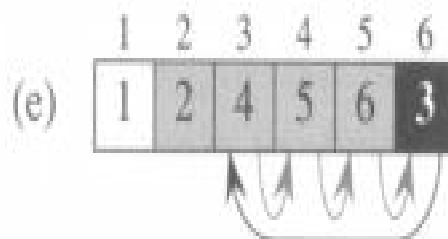
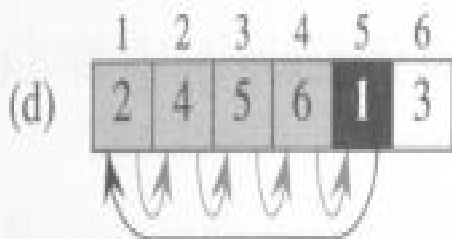
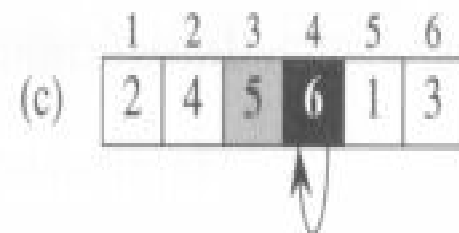
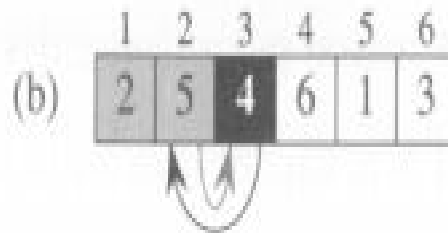
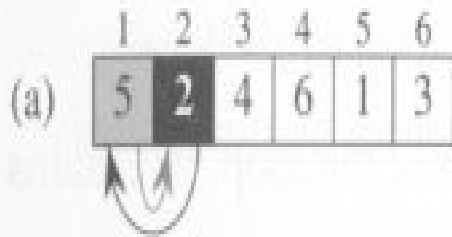
Drugo predavanje

Sortiranje listi

# Sortiranje



# Sortiranje liste



# Algoritam sortiranja umetanjem

1. za  $j \leftarrow 2$  do dužina  $[A]$
2.      $var \leftarrow A[j]$
3.      $\gg\gg$ umetni  $A[j]$  u sekvencu  $A[1 .. j-1]$  $\ll\ll$
4.      $i \leftarrow j - 1$
5.     dok je  $i > 0$  i  $A[i] > var$
6.          $A[i+1] \leftarrow A[i]$
7.          $i \leftarrow i - 1$
8.      $A[i+1] \leftarrow var$

# Invarijante algoritma (postupka)

- ❖ Istinitost prije djelovanja (početka)
  - Ispravnost prvog koraka
  - Trivijalno sortirano
  - Za  $j=2$   $A[1, \dots, j-1]$  ima samo jedan element
- ❖ Ako je postupak valjan prije koraka iteracije valjan je i u idućem koraku (indukcija)
- ❖ Nakon konačnog broja valjanih instrukcija dobivamo ispravan rezultat
  - konačnost petlje

# Inicijalizacija

## ❖ Invarijante algoritma

- Istinitost prije djelovanja
- Trivijalno sortirano
- Za  $j=2$   $A[1, \dots, j-1]$  ima samo jedan element

# Postupak

## ❖ Invarijante algoritma

- vanjska petlja
- unutarnja petlja
- uspoređivanje
- posmak

# Završetak zadatka

❖ Invarijante algoritma

■ Uređena lista



# Analiza zadatka

- |    |   |                                  |
|----|---|----------------------------------|
| 1. | za $j \leftarrow 2$ do dužina $[A]$           | ■ $C_1^*(n-1)$                   |
| 2. | $\text{var} \leftarrow A[j]$                  | ■ $C_2^*(n-1)$                   |
| 3. | $>$ umetni $A[j]$ u sekv. $A[1$<br>$.. j-1]<$ | ■ $0^*(n-1)$                     |
| 4. | $i \leftarrow j-1$                            | ■ $C_4^*(n-1)$                   |
| 5. | dok je $i > 0$ i $A[i] > \text{var}$          | ■ $C_5^* \sum j$ za $j=2..n$     |
| 6. | $A[i+1] \leftarrow A[i]$                      | ■ $C_6^* \sum (j-1)$ za $j=2..n$ |
| 7. | $i \leftarrow i-1$                            | ■ $C_7^* \sum (j-1)$ za $j=2..n$ |
| 8. | $A[i+1] \leftarrow \text{var}$                | ■ $C_8^*(n-1)$                   |

# Zadaća

- ❑ Zbroji sve doprinose iz prethodne analize i izrazi vrijeme potrebno za izvršenje algoritma kao funkciju od  $n$  (broja članova liste koju treba sortirati) u najpovoljnijem i najnepovoljnijem slučaju rasporeda elemenata u listi!
- ❑ Odredi vodeću potenciju u tom razvoju za oba slučaja!
- ❑ Rezultat pošalji na adresu:

[dandroic@phy.hr](mailto:dandroic@phy.hr)

# Funkcija brzine rasta

- Funkcionalnu ovisnost broja postupaka o vrijednosti “n” broja elemenata u listi izražavamo najistaknutijom potencijom od “n” u razvoju analize algoritma
- $\theta(n) \propto n^2$  npr. za naš primjer
  - Najnepovoljniji slučaj
  - Najpovoljniji slučaj
  - Najvjerojatniji slučaj
  - diskusija

# Sortiranje Odabirom Elementa u listi po zadanom kriteriju

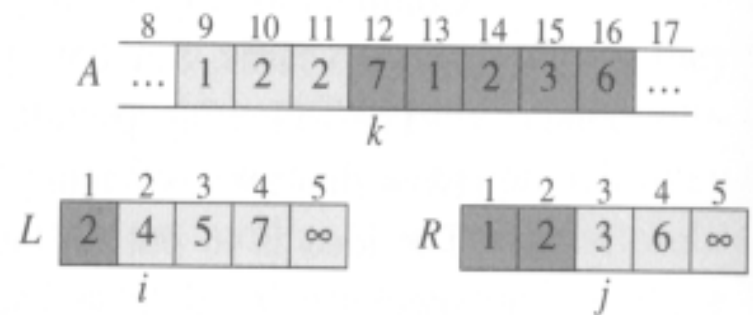
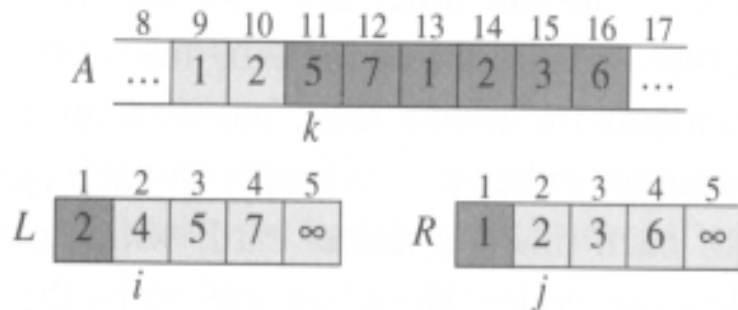
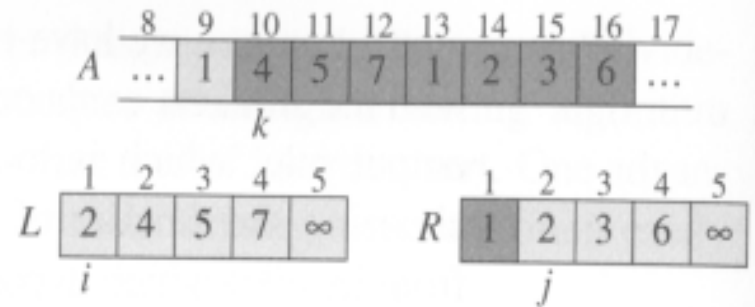
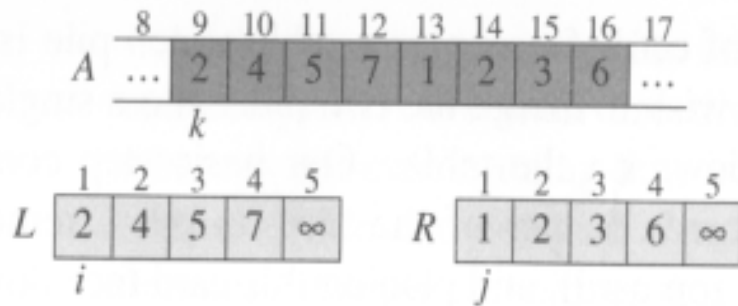
- ❑ Napravi shematski program u apstraktnom kodu i izračunaj efikasnost algoritma sortiranja ako je ideja postupka traženje najvećeg elementa u nesortiranom dijelu (pod) liste
- ❑ Odredi vodeću potenciju u tom razvoju za oba slučaja (povoljniji i nepovoljniji) !
- ❑ Rezultat pošalji na adresu:

[dandroic@phy.hr](mailto:dandroic@phy.hr)

# “Podjeli pa vladaj” princip

- Osnovna karakteristika rekurzivnost
- Podjela problema
- Savladavanje pod problema; ponekad i na najizravniji način
- Ponavljanje postupka
- Rješenje
- Mogućnost algoritmiziranja problema rekurzivnim postupkom i prednosti nad drugim očiglednijim idejama rješavanja problema

# Sortiranje umetanjem listi (Merge Sort a-d)



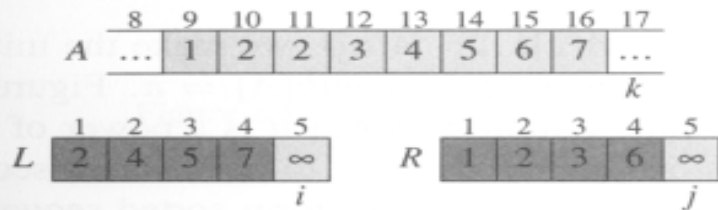
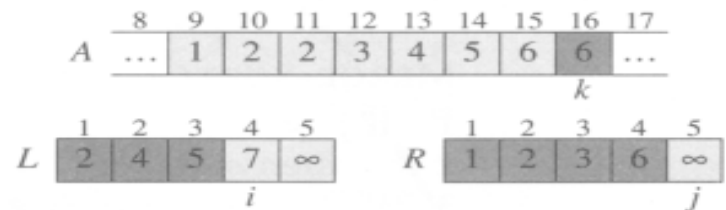
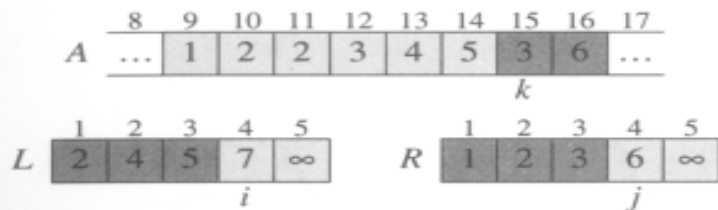
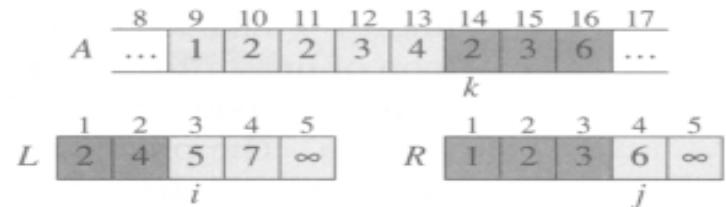
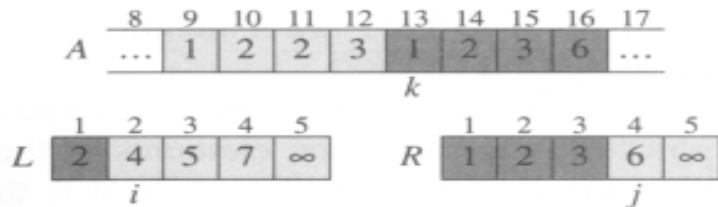
(a)

(b)

(c)

(d)

# Sortiranje umetanjem listi (Merge Sort e-i)



# Sortiranje umetanjem listi

## ■ MERGE( A, p, q, r)

- 1)  $n_1 \leftarrow q - p + 1$
- 2)  $n_2 \leftarrow r - q$
- 3) napravi polja  $L[1, \dots, n_1 + 1]$ ,  
 $R[1, \dots, n_2 + 1]$
- 4) za  $i \leftarrow 1$  do  $n_1$
- 5)      $L[i] \leftarrow A[p + i - 1]$
- 6) za  $j \leftarrow 1$  do  $n_2$
- 7)      $R[j] \leftarrow A[q + j]$
- 8)  $L[n_1 + 1] \leftarrow \infty$
- 9)  $R[n_2 + 1] \leftarrow \infty$
- 10)  $i \leftarrow 1$
- 11)  $j \leftarrow 1$
- 12) za  $k \leftarrow p$  do  $r$
- 13)    ako je  $L[i] \leq R[j]$
- 14)        tada  $A[k] \leftarrow L[i]$
- 15)             $i \leftarrow i + 1$
- 16)        inače  $A[k] \leftarrow R[j]$
- 17)             $j \leftarrow j + 1$



# Analiza (MERGE) algoritma sortiranja umetanjem listi

- ❑ Odredi funkciju  $\theta(n)$  –  
vodeću potenciju broja operacija u  
ovisnosti o broju elemenata u nesortiranoj  
listi  
odredi invarijante algoritma  
diskutiraj probleme
- ❑ Rezultat pošalji na adresu:

[dandroic@phy.hr](mailto:dandroic@phy.hr)