

Príklad 1: Radix triedenie – lexikologické triedenie

Vstup : slová A_1, A_2, \dots, A_n

každé $A_i = (a_{i1}, a_{i2}, \dots, a_{ik})$, $a_{ij} \in \langle 0, m-1 \rangle$

zložitosť $O((m+n)k)$

Výstup: slová B_1, B_2, \dots, B_n

Pričom $B_i \leq B_{i+1}$

Algoritmus:

```
begin
  rad( $A_1, A_2, \dots, A_n$ )
  for  $j \leftarrow k$  step -1 until 1 do
    begin
      for  $l \leftarrow 0$  until  $m-1$  do  $Q[l] = \dots$  slová, ktoré majú  $l$  na  $j$ -tom mieste
      while  $rad \neq 0$  do
        begin
          for  $i \leftarrow 1$  until  $n$  do
             $Q[a_{ij}] \leftarrow A_i$ 
          end
          for  $l \leftarrow 0$  until  $m-1$  do  $rad \leftarrow Q[l]$ 
        end
      end
    end
  end
```

Príklad:

abc, dcb, bca, bbc, acc, bac, baa $A_1 \quad A_2, \quad \dots \quad A_7$	$j = 3$	$Q[a]$ bca baa	$Q[b]$ dcb	$Q[c]$ abc bbc acc bac	$Q[d]$
bca, baa, dcb, abc, bbc, acc, bac	$j = 2$	$Q[a]$ baa bac	$Q[b]$ abc bbc	$Q[c]$ bca dcb acc	$Q[d]$
baa, bac, abc, bbc, bca, dcb, acc	$j = 1$	$Q[a]$ abc acc	$Q[b]$ baa bac bbc bca	$Q[c]$	$Q[d]$ dcb
abc, acc, baa, bac, bbc, bca, dbc $B_1 \quad B_2 \quad \dots \quad B_7$					

Príklad 2: Radix triedenie slov roznej dĺžky.

Vstup : A_1, A_2, \dots, A_n
 l_1, l_2, \dots, l_n

Výstup: B_1, B_2, \dots, B_n , $B_i \leq B_{i+1}, i= 1, \dots, n-1$

Postup: pole $rad (A_1, A_2, \dots, A_n)$
 $Q[a], Q[b], \dots$
 $N[1], \dots, N[l_{\max}]$ symboly na l_i pozícii reťazca
 $D[1], \dots, D[l_{\max}]$ reťazce dĺžky $1, 2, \dots, l_{\max}$.

Algoritmus:

```
begin
  for  $l \leftarrow l_{\max}$  step  $-1$  until  $1$  do
    begin
      pridať obsah  $D[l]$  na začiatok radu
      while  $rad \neq 0$  do
        begin
          vybrať  $A_i$  z radu do  $Q[a_{il}]$ 
        end;
        for  $j \in N[l]$  do
          begin
            vložiť obsah  $Q[j]$  do radu (od začiatku do konca)
            vymazať  $Q[j]$ 
          end
        end
      end
    end
  end.
```

Príklad:

lexikologicky usporiadať:
cc, a, bc, aab, baca, cbc
 A_1, A_2, \dots, A_6
 $l = 2 \ 1 \ 2 \ 3 \ 4 \ 3 \Rightarrow l_{\max} = 4 ; j = a, b, c$

$D[1]$	$D[2]$	$D[3]$	$D[4]$
a	cc	aab	baca
	bc	cbc	

$N[1]$	$N[2]$	$N[3]$	$N[4]$
c	c	b	a
a	a	c	
b	b		

l = 4	rad: baca	Q[a] baca	Q[b]	Q[c]
	rad: baca			
l = 3	rad: aab, cbc, baca	Q[a]	Q[b] aab	Q[c] cbc baca
	rad: aab, cbc, baca			
l = 2	rad: cc, bc, aab, cbc, baca	Q[a] aab baca	Q[b] cbc	Q[c] cc bc
	rad: cc, bc, aab, cbc, baca			
l = 1	rad: a, aab, baca, cbc, cc, bc	Q[a] a aab	Q[b] baca bc	Q[c] cbc cc
	rad: a, aab, baca, cbc, cc, bc			
	B ₁ B ₂	B ₆		

Príklad 3. Heapsort - stromové triedenie

Vstup: A_1, \dots, A_n

Výstup: $A_1 \leq A_{\lfloor n/2 \rfloor}$

Zložitosť $O(n \log n)$

Postup: 1. vytvorenie heapu (-haldy):

```
heap(i,j)
  a) if  $A_i$  nie je list a jeho syn  $A_k > A_i$  then
    begin
  b)  $A_i \leftrightarrow A_k$ 
  c) heap(k,j)
  end
```

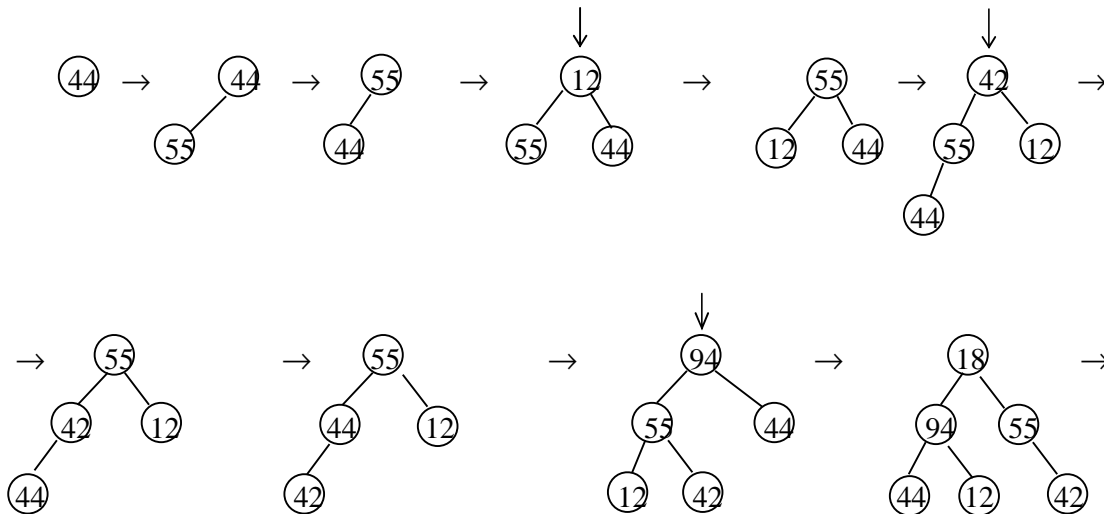
2. heapsort :

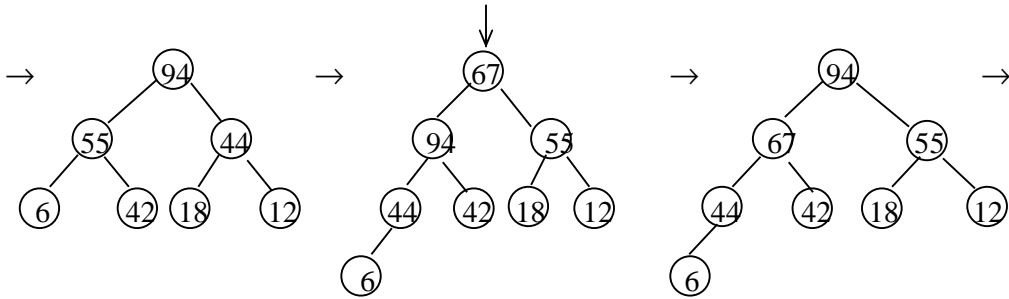
```
begin
  for i ← n step -1 until 1 do heap(i,n)
  for i ← n step -1 until 2 do
    begin
       $A_1 \leftrightarrow A_i$ 
      heap(1, i-1)
    end
  end
```

Príklad: zotriedit čísla : 44, 55, 12, 42, 94, 18, 6, 67

$n=8$

1. vytvorim heap





Príklad 4.

Quicksort - triedenie výmenou

Vstup: $a_1, a_2, \dots, a_n = \text{množina } S$

Výstup: $b_1, b_2, \dots, b_n, b_i \leq b_{i+1}$

Postup: procedure Quickstart(S)

1. if S obsahuje najviac 1 prvok then return S

else begin

2. vyberieme ľubovoľný prvok z S, napr. a

3. rozdelíme $S = S_1 \cup S_2 \cup S_3$; $S_1 = \{ a_i \mid a_i < a \}$

$S_2 = \{ a_i \mid a_i = a \}$

$S_3 = \{ a_i \mid a_i > a \}$

4. Quicksort (S_1)

S_2

5. Quicksort (S_3)

end;

algoritmus rozdelenia S na S_1 a $S_2 \cup S_3$:

$\|S\| = n, i = 1, \dots, n$

$1 \leq f \leq l \leq n$

$S_1 = \{ a[f], a[f+1], \dots, a[k] \}$

$S_2 \cup S_3 = \{ a[k+1], \dots, a[l] \}$

$f \leq k \leq l$

begin

$i \leftarrow f$

$j \leftarrow l$

while $i \leq j$ do

begin

while $a[j] \geq a$ and $j \geq f$ do $j \leftarrow j - 1$;

while $a[j] < a$ and $i \leq l$ do $i \leftarrow i + 1$;

if $i < j$ then

begin

$a[i] \leftrightarrow a[j]$

$i \leftarrow i + 1$

$j \leftarrow j - 1$

end

end

end

Príklad: zotried'te quicksortom čísla

38, 12, 6, 41, 17, 29, 23, 8, 68

a_1, \dots, a_9

vyberieme $a = 29$

1. $i = 1$ 38 nie je menšie než 29

$j = 9$ 68 > 29 ok

$j = 8$ 29 nie je menšie než 8

vymeníme 8, 38 : 8, 12, 6, 41, 17, 29, 23, 38, 68

2. $i = 2$: 12 < 29 ok

$j = 7$: 29 nie je menšie než 23

$i = 3$: 6 < 29 ok

i = 4 : 41 nie je menšie než 29
vymeníme 23, 41 : 8, 12, 6, 23, 17, 29, 41, 38, 68

3. i = 5 : 17 < 29
j = 6 : 29 nie je menšie než 29
i = 6 : 29 nie je menšie než 29
8, 12, 6, 23, 17, 29, 41, 38, 68
 $\{ 8, 12, 6, 23, 17 \} = S_1$
 $\{ 29, 41, 38, 68 \} = S_2 \cup S_3$

Quicksort (S_1): podľa algoritmu rekurzívne zotriedi :

6, 8, 12, 17, 23

Quicksort ($S_2 \cup S_3$):

29, 38, 41, 68

Bubble sort - triedenie priamou výmenou

Postupne sa porovnávajú a vymieňajú dvojice susedných prvkov až po zotriedenie. Cyklicky sa spracúva pole $a[1 \dots n]$. V každom prechode sa vyberie najmenší prvok skúmanej podmnožiny, ktorý "vybude" nahor.

algorithmus:

```
procedure bubblesort;
```

$$\underline{\text{var}} \ i, j : \text{index}; x : \text{položka};$$
begin

for i := 2 to n do

for j := n step -1 until i do

if $a[j - 1] > a[j]$ then

begin

```

x := a[j-1] ;

```

$$a[j-1] := a[j]$$
$$a[j] := x$$

výmena

endend

Príklad: zotriediť bubblesortom čísla 48, 15, 42, 12, 8, 64, 18, 1

The diagram illustrates the movement of the value 1 through a 7x7 grid over 7 iterations. Red arrows show the path of the value 1, which moves right and then up at each step.

	i = 2	i = 3	i = 4	i = 5	i = 6	i = 7
48	1	1	1	1	1	1
15	48	8	8	8	8	8
42	15	48	12	12	12	12
12	42	15	48	15	15	15
8	12	42	15	48	18	18
64	8	12	42	18	48	42
18	64	18	18	42	42	48
1	18	64	64	64	64	64

Príklad číslo 6.

Bubble sort with shake - s pretriasaním

- strieda smer prehl'adávania v jednotlivých po sebe idúcich prechodoch.
- algoritmus končí, ak v poslednom prechode nenastane výmena.

Príklad: Zotriedime čísla 48, 15, 42, 12, 8, 64, 18, 1

<l, r> - interval prehl'adávania

