

# Triedenie

# Problém

Dané pole prirodzených čísel utriediť vzostupne.

Vstup: 5, 3, 7, 10, 2

Výstup: 2, 3, 5, 7, 10

# Pomalé triedenie

pole  $x[0], x[1], \dots, x[n-1]$

```
for(i=0;i<n-1;++i)
  for(j=i+1;j<n;++j)
    if (x[j]<x[i]) swap(&x[i],&x[j]);
```

# Pomalé triedenie - zložitosť

Koľkokrát sa porovnáva ( $x[j] < x[i]$ )?

$$T(n) = \sum_{i=0}^{n-2} \sum_{j=i+1}^{n-1} 1$$

$$T(n) \sim n^2/2$$

$$T(n) = \theta(n^2)$$

# Pomalé triedenie - zložitosť

- najhorší prípad?
- najlepší prípad?
- priemerná zložitosť?

# Quicksort - úvod

Pole 16,8,10,4,25,35,30,59,25

všimnime si piaty prvok zľava, t.j. 25

prvky naľavo od neho sú menšie než 25

prvky napravo od neho sú väčšie alebo rovné než 25

⇒ piaty prvok tohto poľa zostane v usporiadanom poli na tomto mieste

Pivot

# Pivot

## Definition

Číslo  $T$  postupnosti  $x[0], x[1], \dots, x[r]=T, x[r+1], \dots, x[n-1]$  sa volá pivot, ak

- $x[i] < T \quad \forall i, 0 \leq i < r,$
- $x[i] \geq T \quad \forall i, r < i < n.$

# Quicksort - myšlienka

- vytvoríť v poli pivota
- rekurzívne to opakovať v podpoliach naľavo, resp. napravo od pivota



# Ako vytvoríť pivota

```
int split(int *x, int left, int right)
{
0   int j,L,T;
1   L ← náhodne vybraný prvok z intervalu <left,right>;
2   swap(x+left,x+L);
3   T=x[left];i=left;
4   for(j=left+1,j<=right,++j)
5       if x[j]<T {++i; swap(x+i,x+j);}
6   swap(x[left],x[i]);
7   return L;
}
```

## Split - príklad

	3	5	7	1	2
$i = 1, L = 2, T = 5$					
$j = 2$	5	3	7	1	2
$i = 2, j = 3$	5	3	7	1	2
$i = 2, j = 5$	5	3	7	1	2
$i = 3, j = 5$	5	3	1	7	2
$i = 4$	2	3	1	5	7

# Správnošť funkcie split

## Theorem

*Funkcia split usporiada pole  $x$  tak, že  $x[i]=T$  je pivotom poľa  $x$ .*

Dôkaz indukciou:

po každom kroku 5,6,7 pre  $j = left + 1 \dots right$  platí:

- ❶  $x[left] = T$
- ❷ pre  $left < r \leq i$  platí  $x[r] < T$
- ❸ pre  $i < r \leq j$  platí  $T \leq x[r]$

# Quicksort

```
void quicksort(int *x, int left, int right)
{
    int L;
    L=split(x,left,right);
    quicksort(x,left,L-1);
    quicksort(x,L+1,right);
}

quicksort(x,0,n-1);
```

# Quicksort - zložitosť

najhorší prípad:

Split vyberie vždy najmenší (alebo najväčší prvok poľa)

$$f(n) = (n - 1) + f(n - 1) = (n - 1) + (n - 2) + \dots + 1 = \frac{n(n-1)}{2}$$

$$f(n) = \theta(n^2)$$

# Quicksort - zložitosť

priemerne:

$$F(n) = (n - 1) + \frac{1}{n} \sum_{i=1}^n (F(i - 1) + F(n - i))$$

po úprave:

$$F(n) = \theta(n \log n), \text{ resp. } F(n) \sim 2n \log n$$