

Pikajärjestäminen (Quick Sort)

COMP.CS.300 Tietorakenteet ja algoritmit 1

Matti Rintala (matti.rintala@tuni.fi)

Pikajärjestäminen (Quick Sort)

- Hajoita ja hallitse -algoritmi
- Hajoita:
 - Pilkotaan data "pieniin" ja "suuriin"
 - (Ideaalitapauksessa palat yhtä suuria)
- Rekursiivinen hallitse:
 - Järjestetään palat rekursiivisesti samalla algoritmilla
 - 1:n mittaista ei enää tarvitse järjestää

Quicksort(A , $left$, $right$)

- 1 **if** $left < right$ **then** (triviaalitapaukselle ei tehdä mitään)
- 2 $pivot := \text{Partition}(A, left, right)$ (jaetaan pieniin ja suuriin, $pivot$ jakokohta)
- 3 Quicksort(A , $left$, $pivot-1$) (järjestetään jakoalkiota pienemmät)
- 4 Quicksort(A , $pivot+1$, $right$) (järjestetään jakoalkiota suuremmat)

| | | | | | | | | |
|----|---|---|---|---|---|----|---|---|
| 10 | 4 | 2 | 6 | 9 | 3 | 14 | 8 | 5 |
|----|---|---|---|---|---|----|---|---|

| | | | | | | | | |
|---|---|---|---|---|----|----|---|---|
| 4 | 2 | 3 | 5 | 9 | 10 | 14 | 8 | 6 |
|---|---|---|---|---|----|----|---|---|

| | | | | | | | | |
|---|---|---|---|---|----|----|---|---|
| 2 | 3 | 4 | 5 | 9 | 10 | 14 | 8 | 6 |
|---|---|---|---|---|----|----|---|---|

| | | | | | | | | |
|---|---|---|---|---|----|----|---|---|
| 2 | 3 | 4 | 5 | 6 | 10 | 14 | 8 | 9 |
|---|---|---|---|---|----|----|---|---|

| | | | | | | | | |
|---|---|---|---|---|---|---|----|----|
| 2 | 3 | 4 | 5 | 6 | 8 | 9 | 10 | 14 |
|---|---|---|---|---|---|---|----|----|

| | | | | | | | | |
|---|---|---|---|---|---|---|----|----|
| 2 | 3 | 4 | 5 | 6 | 8 | 9 | 10 | 14 |
|---|---|---|---|---|---|---|----|----|

| | | | | | | | | |
|----|---|---|---|---|---|----|---|---|
| 10 | 4 | 2 | 6 | 9 | 3 | 14 | 8 | 5 |
|----|---|---|---|---|---|----|---|---|

| | | | | | | | | |
|---|---|---|---|---|----|----|---|---|
| 4 | 2 | 3 | 5 | 9 | 10 | 14 | 8 | 6 |
|---|---|---|---|---|----|----|---|---|

| | | | | | | | | |
|---|---|---|---|---|----|----|---|---|
| 2 | 3 | 4 | 5 | 9 | 10 | 14 | 8 | 6 |
|---|---|---|---|---|----|----|---|---|

| | | | | | | | | |
|---|---|---|---|---|----|----|---|---|
| 2 | 3 | 4 | 5 | 6 | 10 | 14 | 8 | 9 |
|---|---|---|---|---|----|----|---|---|

| | | | | | | | | |
|---|---|---|---|---|---|---|----|----|
| 2 | 3 | 4 | 5 | 6 | 8 | 9 | 10 | 14 |
|---|---|---|---|---|---|---|----|----|

| | | | | | | | | |
|---|---|---|---|---|---|---|----|----|
| 2 | 3 | 4 | 5 | 6 | 8 | 9 | 10 | 14 |
|---|---|---|---|---|---|---|----|----|

Pikajärjestäminen: Ositus (Partition)

COMP.CS.300 Tietorakenteet ja algoritmit 1

Matti Rintala (matti.rintala@tuni.fi)

Partition(*A*, *left*, *right*)

```
1 pivot := A[right]           (otetaan pivotiksi viimeinen alkio)
2 cut := left - 1             (merkitään cut:lla pienten puolen loppua)
3 for i := left to right-1 do (käydään läpi toiseksi viimeiseen alkioon asti)
4   if A[i] ≤ pivot          (jos A[i] kuuluu pienten puolelle...)
5     cut := cut + 1          (... kasvatetaan pienten puolta...)
6     A[cut] ⇌ A[i]          (... ja siirretään A[i] sinne)
7 A[ cut+1 ] ⇌ A[ right ]    (sijoitetaan pivot pienten ja isojen puolten väliin)
8 return cut+1              (palautetaan pivot-alkion uusi sijainti)
```

| | | | | | | | | |
|----|---|---|---|---|---|----|---|---|
| 10 | 4 | 2 | 6 | 9 | 3 | 14 | 8 | 5 |
|----|---|---|---|---|---|----|---|---|