

Quicksort

Weiss, avsnitt 7.7

Peter Ljunglöf

DATo36, Datastrukturer

30 nov 2012

Quicksort

Quicksort väljer ett specifikt värde (kallat *pivot*), och delar upp resten av fältet i två delar:

- alla element som är $\leq$ pivot läggs i vänstra delen
- alla element som är $>$ pivot läggs i högra delen
- pivoten placeras mittemellan
- sortera vänstra och högra delen var för sig, rekursivt

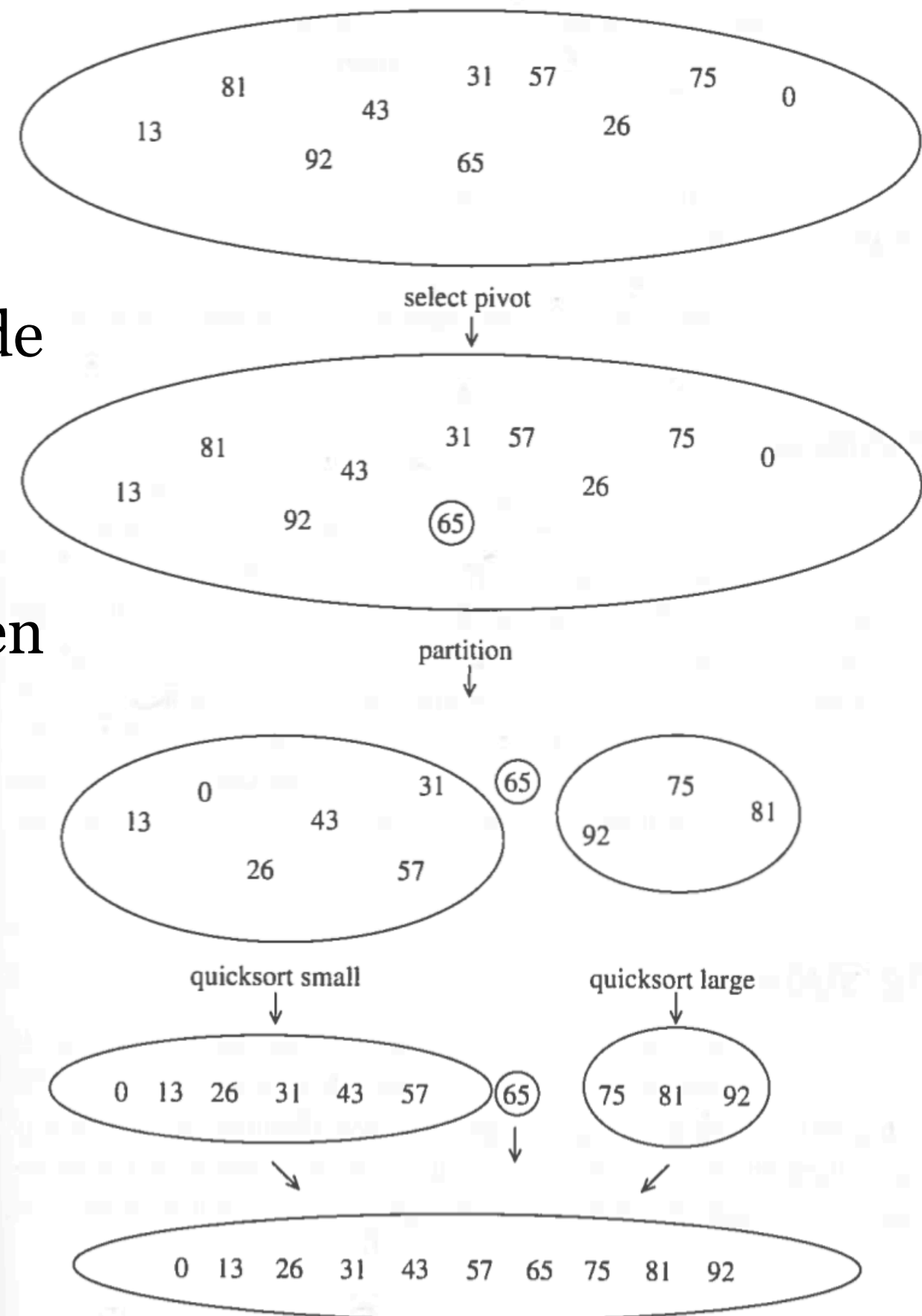


Figure 7.11 The steps of quicksort illustrated by example

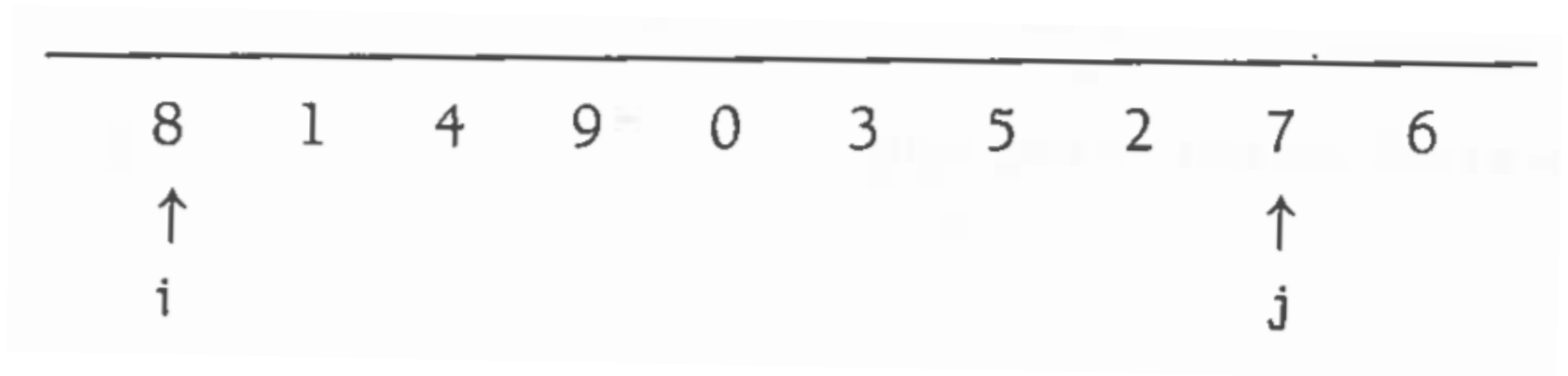
Analys av Quicksort

Om pivoten är ett slumpmässigt värde från det aktuella fältet,

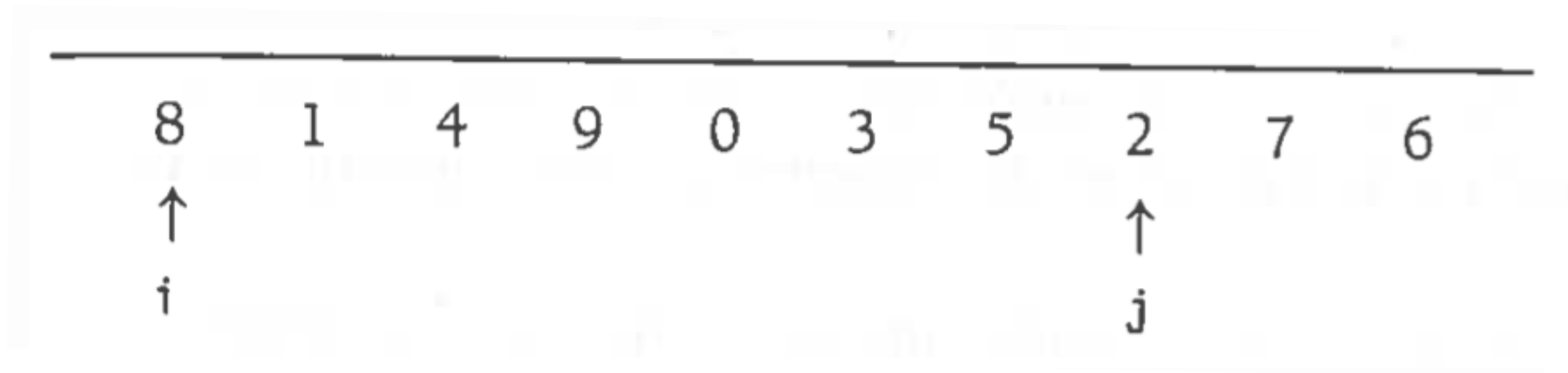
- så kommer statistiskt sett pivoten att ligga i mitten av fältet
- vänster och höger halva kommer (statistiskt sett) att bli lika stora
- dvs, varje rekursivt anrop kommer att sortera ett hälften så stort fält (statistiskt sett)
- dvs, samma resonemang som för Mergesort
- fast Quicksort är in-place

Partitionering

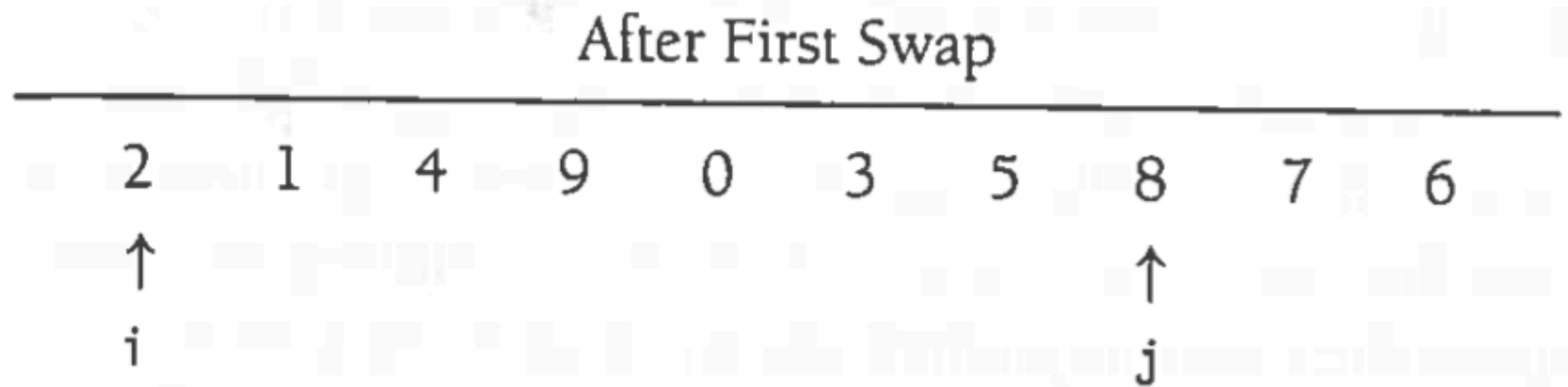
Partitioning



Partitioning



Partitioning



Partitioning

Before Second Swap

2	1	4	9	0	3	5	8	7	6
			↑			↑			
			i			j			

Partitioning

After Second Swap

2	1	4	5	0	3	9	8	7	6
			↑			↑			
			i			j			

Partitioning

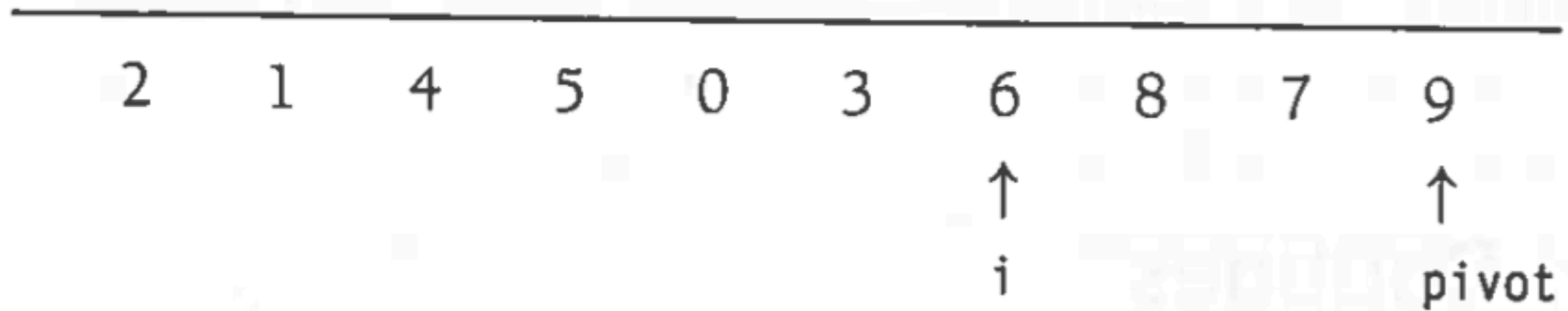
Before Third Swap

2 1 4 5 0 3 9 8 7 6

↑ ↑
j i

Partitioning

After Swap with Pivot



Analys av Quicksort

Om pivoten väljs dumt så kan Quicksort bli ineffektiv:

- om det ena delfältet alltid råkar bli tomt
- t.ex., om man alltid tar första värdet som pivot, och listan redan är sorterad
- i det fallet blir komplexiteten $O(n^2)$, precis som insättningssortering

Bättre är att välja pivoten smartare:

- t.ex., utgå från tre värden — första, mittersta och sista elementet
- välj medianen av dessa som pivot
- detta minskar risken för dålig delning