

# Quick sort using recursive method

- Recursive Algorithm
  - Example
- Quick sort complexity

# Recursive Algorithm cont...

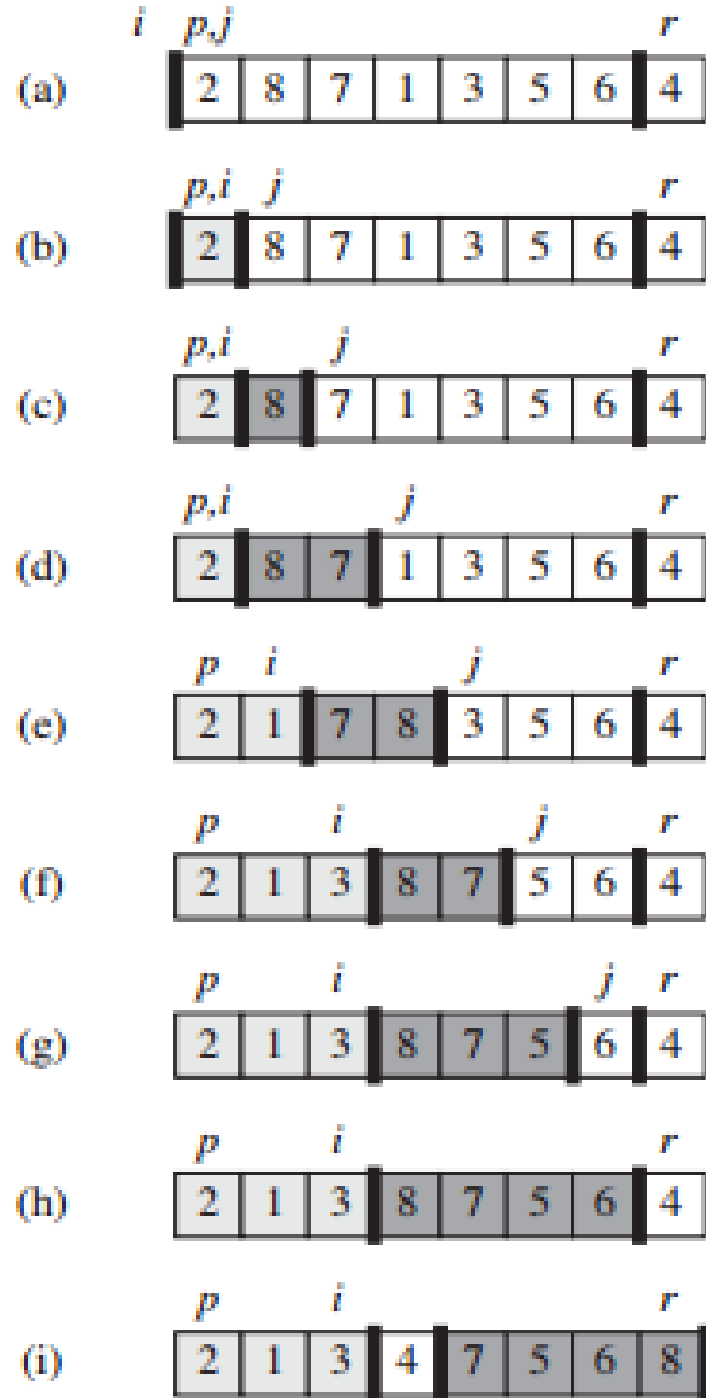
- QUICKSORT(A, p, r): A: array, p: LB, r=UB
  1. **if**  $p < r$
  2.      $q = \text{PARTITION}(A, p, r)$
  3.     QUICKSORT(A, p,  $q-1$ )
  4.     QUICKSORT(A,  $q + 1$ , r)
  5. Exit

# Recursive Algorithm(partition)

PARTITION(A, p, r)

1.  $x = A[r]$  // pivot is last value
2.  $i = p - 1$
3. Repeat steps 4 to 6 **for**  $j = p$  **to**  $r - 1$
4.     **if**  $A[j] \leq x$
5.          $i = i + 1$
6.         exchange  $A[i]$  with  $A[j]$
7. exchange  $A[i + 1]$  with  $A[r]$
8. **return**  $i + 1$

# Example:



# Complexity of Quick sort

- Worst case
  - $O(n^2)$
- Average case
  - $O(n \log n)$
- Best case
  - $O(n \log n)$

Thank You