CIS 263 Introduction to Data Structures and Algorithms

Quick Sort

Quick sort

Recursion:

QUICKSORT(A, p, r)

1 **if** $p < r$
2      $q =$ PARTITION(A, p, r)
3   QUICKSORT($A, p, q - 1$)
4   QUICKSORT($A, q + 1, r$)

PARTITION(A, p, r)

1 $x = A[r]$
2 $i = p - 1$
3 **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** $i + 1$

Quick sort

Recursion:

QUICKSORT(A, p, r)

1 **if** $p < r$
2      $q = \text{PARTITION}(A, p, r)$
3   QUICKSORT$(A, p, q - 1)$
4   QUICKSORT$(A, q + 1, r)$

PARTITION(A, p, r)

1 $x = A[r]$
2 $i = p - 1$
3 **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** $i + 1$
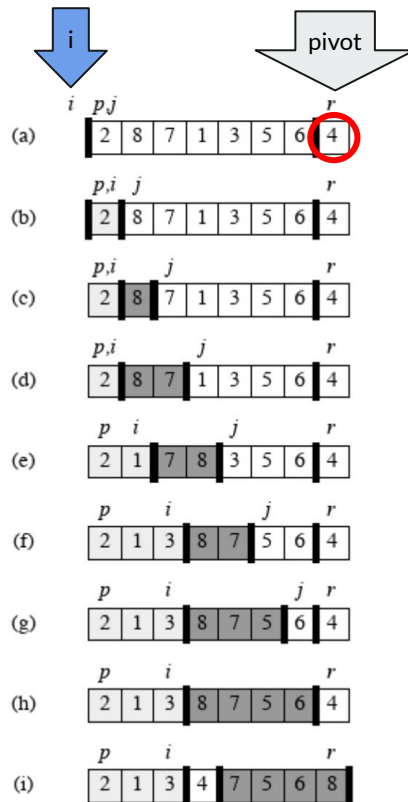
Quick sort

Pivot: r(4), finding its expected position

PARTITION(A, p, r)

1 $x = A[r]$
2 $i = p - 1$
3 **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** $i + 1$

i = -1,
if array index starts at 0

i

pivot

(a) i p,j ... r
2 8 7 1 3 5 6 4

(b) p,i j ... r
2 8 7 1 3 5 6 4

(c) p,i j ... r
2 8 7 1 3 5 6 4

(d) p,i j ... r
2 8 7 1 3 5 6 4

(e) p i j r
2 1 7 8 3 5 6 4

(f) p i j r
2 1 3 8 7 5 6 4

(g) p i j r
2 1 3 8 7 5 6 4

(h) p i r
2 1 3 8 7 5 6 4

(i) p i r
2 1 3 4 7 5 6 8

Quick sort

Pivot: r(4), finding its expected position

PARTITION(A, p, r)
1 x = A[r]
2 i = p − 1
3 **for** j = p **to** r − 1
4   **if** A[j] ≤ x
5   i = i + 1
6   exchange A[i] with A[j]
7 exchange A[i + 1] with A[r]
8 **return** i + 1

i = -1,
if array index starts at 0
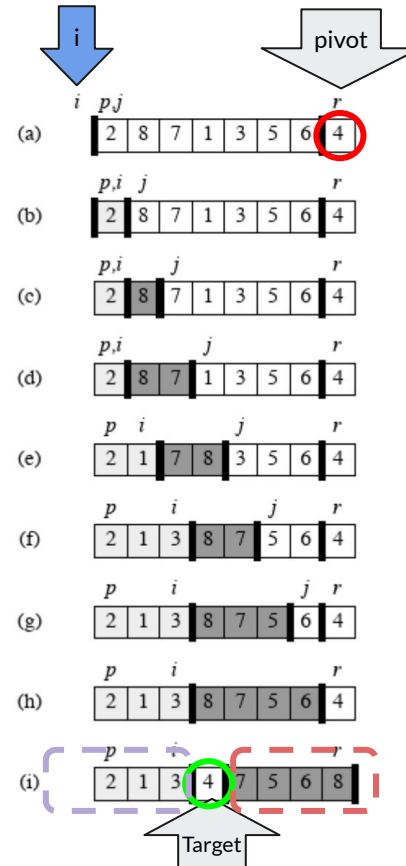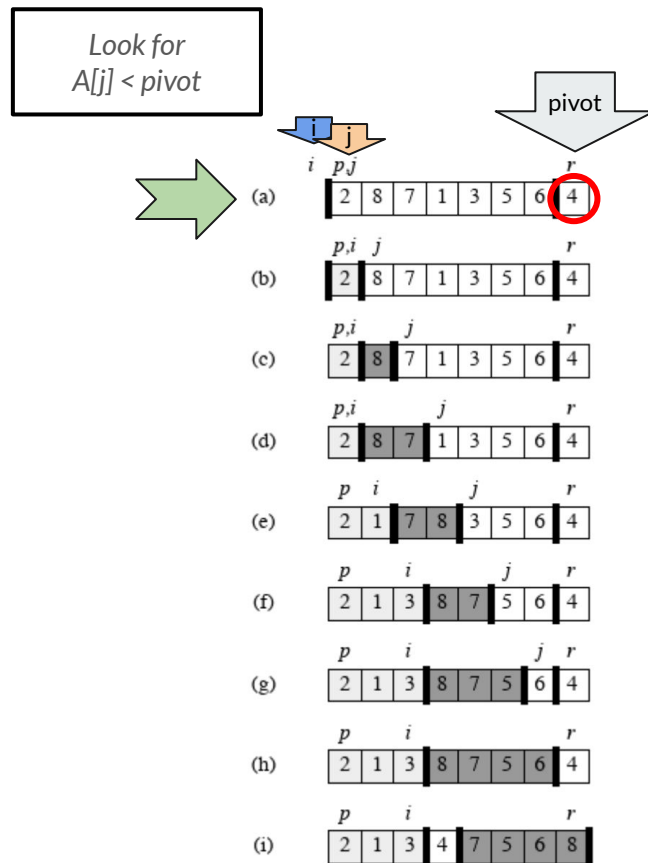

i | pivot

(a) | i p,j   r
    | 2 | 8 | 7 | 1 | 3 | 5 | 6 | 4

(b) | p,i j   r
    | 2 | 8 | 7 | 1 | 3 | 5 | 6 | 4

(c) | p,i   j   r
    | 2 | 8 | 7 | 1 | 3 | 5 | 6 | 4

(d) | p,i   j   r
    | 2 | 8 | 7 | 1 | 3 | 5 | 6 | 4

(e) | p   i   j   r
    | 2 | 1 | 7 | 8 | 3 | 5 | 6 | 4

(f) | p   i   j   r
    | 2 | 1 | 3 | 8 | 7 | 5 | 6 | 4

(g) | p   i   j r
    | 2 | 1 | 3 | 8 | 7 | 5 | 6 | 4

(h) | p   i   r
    | 2 | 1 | 3 | 8 | 7 | 5 | 6 | 4

(i) | p   i   r
    | 2 | 1 | 3 | 4 | 7 | 5 | 6 | 8

Target

Quick sort

Pivot: r(4), finding its expected position

```
PARTITION(A, p, r)
1   x = A[r]
2   i = p − 1
3   for j = p to r − 1
4       if A[j] ≤ x
5           i = i + 1
6           exchange A[i] with A[j]
7   exchange A[i + 1] with A[r]
8   return i + 1
```
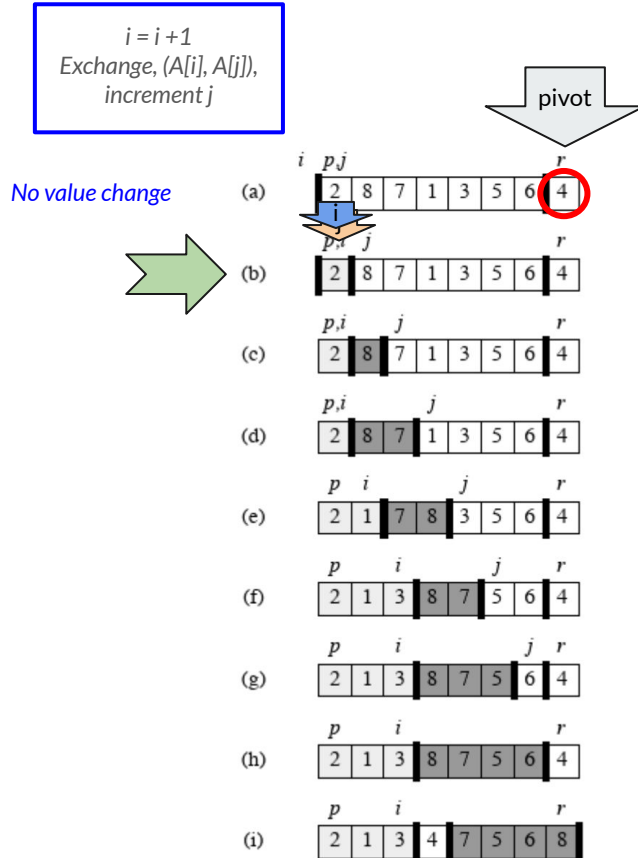
Look for A[j] < pivot

pivot

	i	p,j						r
(a)	2	8	7	1	3	5	6	4

	p,i	j						r
(b)	2	8	7	1	3	5	6	4

	p,i		j					r
(c)	2	8	7	1	3	5	6	4

	p,i			j				r
(d)	2	8	7	1	3	5	6	4

	p		i		j			r
(e)	2	1	7	8	3	5	6	4

	p		i			j		r
(f)	2	1	3	8	7	5	6	4

	p			i			j	r
(g)	2	1	3	8	7	5	6	4

	p			i				r
(h)	2	1	3	8	7	5	6	4

	p			i				r
(i)	2	1	3	4	7	5	6	8

Quick sort

Pivot: r(4), finding its expected position

PARTITION(A, p, r)
1 $x = A[r]$
2 $i = p - 1$
3 **for** $j = p$ **to** $r - 1$
4   **if** $A[j] \leq x$
5          $i = i + 1$
6   exchange $A[i]$ with $A[j]$
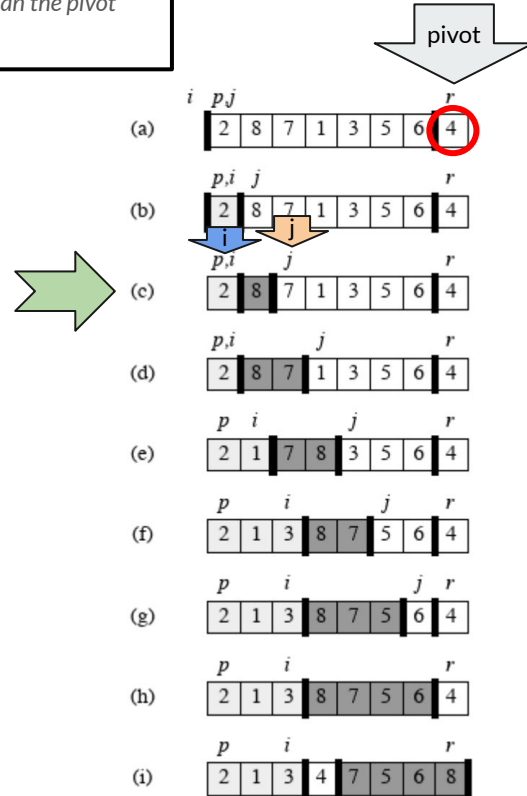7 exchange $A[i + 1]$ with $A[r]$
8 **return** $i + 1$

$i = i + 1$
Exchange, (A[i], A[j]),
increment j

pivot

No value change

(a) | 2 | 8 | 7 | 1 | 3 | 5 | 6 | 4 |
    i p,j   r

(b) | 2 | 8 | 7 | 1 | 3 | 5 | 6 | 4 |
   p,i j   r

(c) | 2 | 8 | 7 | 1 | 3 | 5 | 6 | 4 |
   p,i   j   r

(d) | 2 | 8 | 7 | 1 | 3 | 5 | 6 | 4 |
   p,i   j   r

(e) | 2 | 1 | 7 | 8 | 3 | 5 | 6 | 4 |
  p i   j   r

(f) | 2 | 1 | 3 | 8 | 7 | 5 | 6 | 4 |
  p   i   j   r

(g) | 2 | 1 | 3 | 8 | 7 | 5 | 6 | 4 |
  p   i   j r

(h) | 2 | 1 | 3 | 8 | 7 | 5 | 6 | 4 |
  p   i   r

(i) | 2 | 1 | 3 | 4 | 7 | 5 | 6 | 8 |
  p   i   r

Quick sort

Pivot: r(4), finding its expected position

PARTITION(A, p, r)

1 $x = A[r]$
2 $i = p - 1$
3 **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** $i + 1$
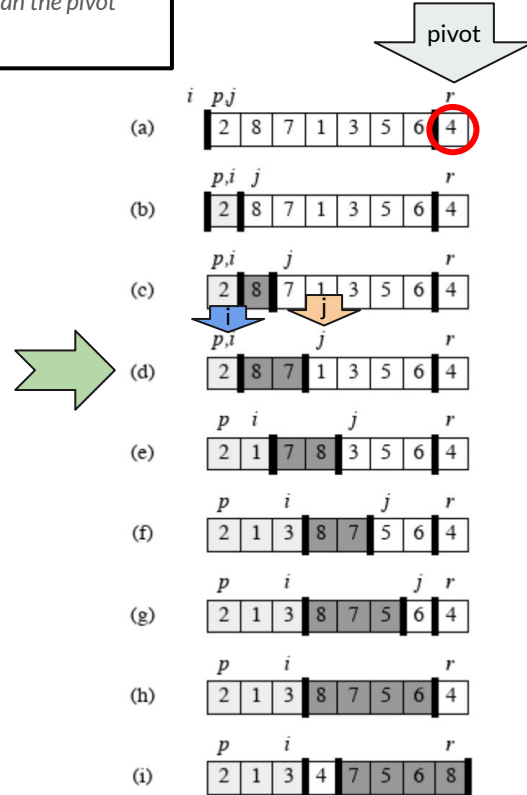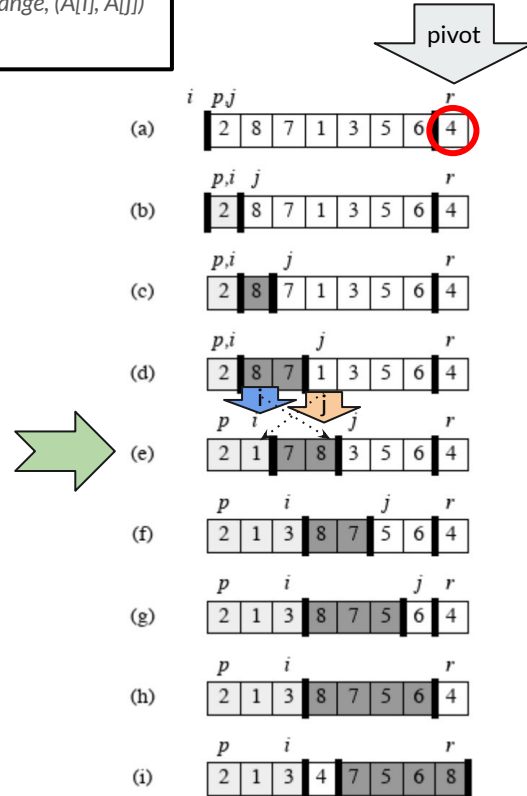
Find the next one less than the pivot

pivot

Quick sort

Pivot: r(4), finding its expected position

$\text{PARTITION}(A, p, r)$

```
1   x = A[r]
2   i = p − 1
3   for j = p to r − 1
4       if A[j] ≤ x
5           i = i + 1
6           exchange A[i] with A[j]
7   exchange A[i + 1] with A[r]
8   return i + 1
```

Find the next one less than the pivot

pivot

Quick sort

Pivot: r(4), finding its expected position

PARTITION(A, p, r)

1 $x = A[r]$
2 $i = p - 1$
3 **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 $A[r]$
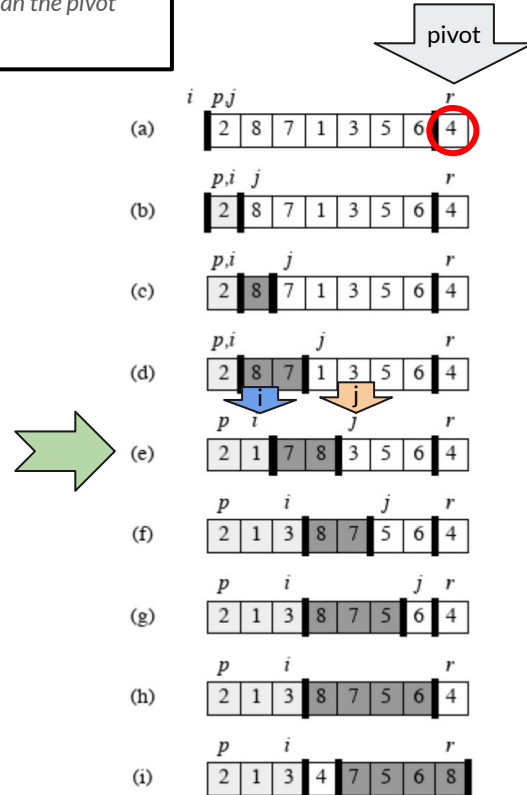8 **return** $i + 1$

i = i +1
Exchange, (A[i], A[j])

pivot

(a) | 2 | 8 | 7 | 1 | 3 | 5 | 6 | 4 |

(b) | 2 | 8 | 7 | 1 | 3 | 5 | 6 | 4 |

(c) | 2 | 8 | 7 | 1 | 3 | 5 | 6 | 4 |

(d) | 2 | 8 | 7 | 1 | 3 | 5 | 6 | 4 |

(e) | 2 | 1 | 7 | 8 | 3 | 5 | 6 | 4 |

(f) | 2 | 1 | 3 | 8 | 7 | 5 | 6 | 4 |

(g) | 2 | 1 | 3 | 8 | 7 | 5 | 6 | 4 |

(h) | 2 | 1 | 3 | 8 | 7 | 5 | 6 | 4 |

(i) | 2 | 1 | 3 | 4 | 7 | 5 | 6 | 8 |

Quick sort

Pivot: r(4), finding its expected position

```
PARTITION(A, p, r)
1   x = A[r]
2   i = p − 1
3   for j = p to r − 1
4       if A[j] ≤ x
5           i = i + 1
6           exchange A[i] with A[j]
7   exchange A[i + 1] with A[r]
8   return i + 1
```

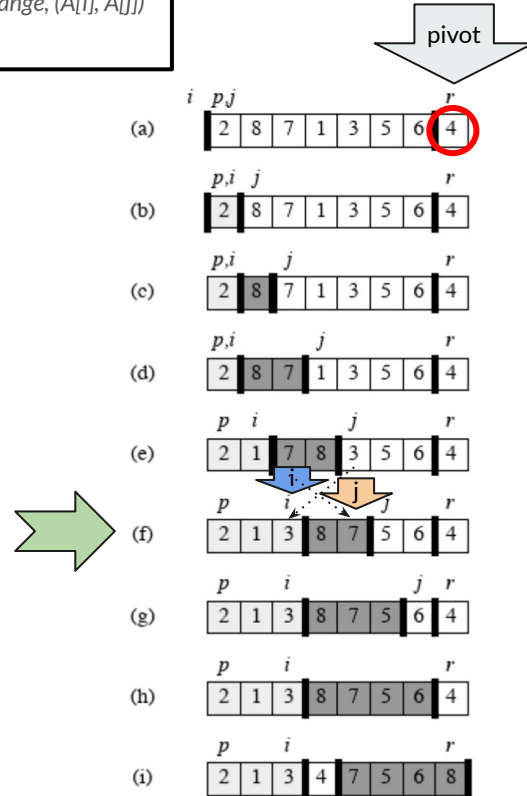*Find the next one less than the pivot*

pivot

Quick sort

Pivot: r(4), finding its expected position

$$\text{PARTITION}(A, p, r)$$

1    $x = A[r]$
2    $i = p - 1$
3   **for** $j = p$ **to** $r - 1$
4   **if** $A[j] \le x$
5           $i = i + 1$
6   exchange $A[i]$ with $A[j]$
7   exchange $A[i + 1]$ with $A[r]$
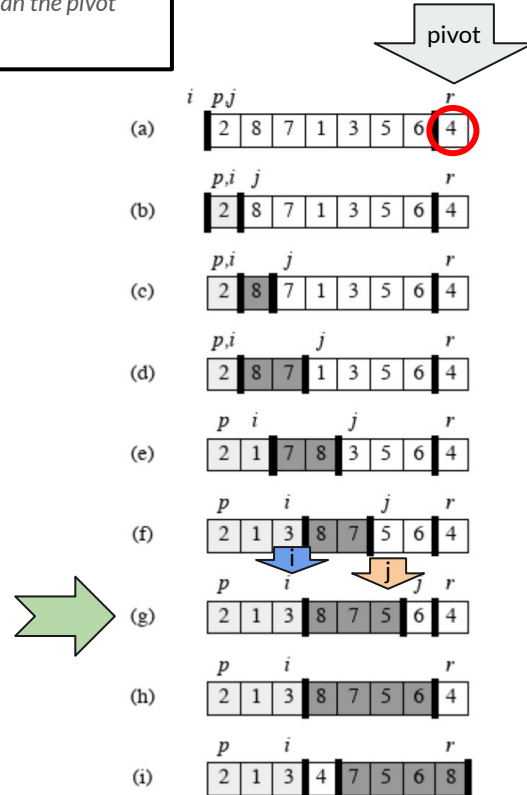8   **return** $i + 1$

$i = i + 1$
Exchange, (A[i], A[j])

pivot

(a) | 2 | 8 | 7 | 1 | 3 | 5 | 6 | 4 |

(b) | 2 | 8 | 7 | 1 | 3 | 5 | 6 | 4 |

(c) | 2 | 8 | 7 | 1 | 3 | 5 | 6 | 4 |

(d) | 2 | 8 | 7 | 1 | 3 | 5 | 6 | 4 |

(e) | 2 | 1 | 7 | 8 | 3 | 5 | 6 | 4 |

(f) | 2 | 1 | 3 | 8 | 7 | 5 | 6 | 4 |

(g) | 2 | 1 | 3 | 8 | 7 | 5 | 6 | 4 |

(h) | 2 | 1 | 3 | 8 | 7 | 5 | 6 | 4 |

(i) | 2 | 1 | 3 | 4 | 7 | 5 | 6 | 8 |

Quick sort

Pivot: r(4), finding its expected position

```
PARTITION(A, p, r)
1   x = A[r]
2   i = p − 1
3   for j = p to r − 1
4       if A[j] ≤ x
5           i = i + 1
6           exchange A[i] with A[j]
7   exchange A[i + 1] with A[r]
8   return i + 1
```
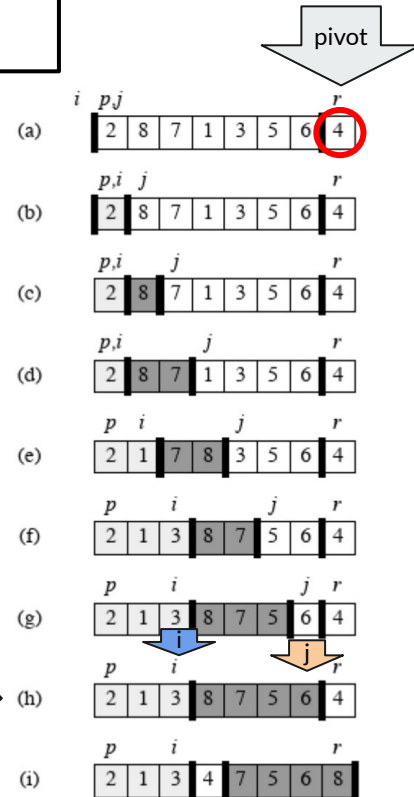
Find the next one less than the pivot

pivot

Quick sort

Pivot: r(4), finding its expected position

PARTITION(A, p, r)

1 $x = A[r]$
2 $i = p - 1$
3 **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 $A[r]$
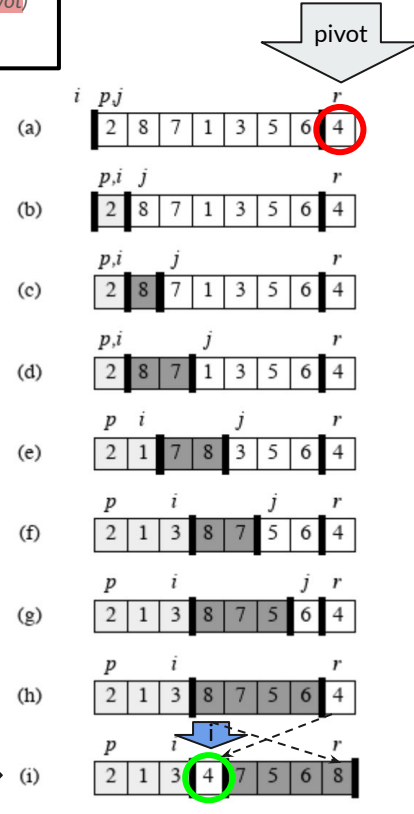8 **return** $i + 1$

We are at r -1

pivot

(a) i p,j ... r | 2 | 8 | 7 | 1 | 3 | 5 | 6 | 4 |

(b) p,i j ... r | 2 | 8 | 7 | 1 | 3 | 5 | 6 | 4 |

(c) p,i j ... r | 2 | 8 | 7 | 1 | 3 | 5 | 6 | 4 |

(d) p,i j ... r | 2 | 8 | 7 | 1 | 3 | 5 | 6 | 4 |

(e) p i j ... r | 2 | 1 | 7 | 8 | 3 | 5 | 6 | 4 |

(f) p i j ... r | 2 | 1 | 3 | 8 | 7 | 5 | 6 | 4 |

(g) p i j r | 2 | 1 | 3 | 8 | 7 | 5 | 6 | 4 |

(h) p i ... r | 2 | 1 | 3 | 8 | 7 | 5 | 6 | 4 |

(i) p i ... r | 2 | 1 | 3 | 4 | 7 | 5 | 6 | 8 |

Quick sort

Pivot: r(4), finding its expected position

PARTITION(A, p, r)

1 $x = A[r]$
2 $i = p - 1$
3 **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** $i + 1$

$i = i + 1$
Exchange, (A[i], pivot)

pivot

(a) | i p,j | ... | r |
2 8 7 1 3 5 6 4

(b) p,i j   r
2 8 7 1 3 5 6 4

(c) p,i   j   r
2 8 7 1 3 5 6 4

(d) p,i   j   r
2 8 7 1 3 5 6 4

(e) p   i   j   r
2 1 7 8 3 5 6 4

(f) p   i   j   r
2 1 3 8 7 5 6 4

(g) p   i   j r
2 1 3 8 7 5 6 4

(h) p   i   r
2 1 3 8 7 5 6 4

(i) p   i   r
2 1 3 4 7 5 6 8

Quick sort

Recursion:

QUICKSORT(A, p, r)

1 **if** $p < r$
2      $q = $ PARTITION(A, p, r)
3   QUICKSORT($A, p, q-1$)
4   QUICKSORT($A, q+1, r$)

PARTITION(A, p, r)

1 $x = A[r]$
2 $i = p - 1$
3 **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** $i + 1$

Quick sort Time Complexity

Average case: **O(n log n)**

Worst case: **O (n)** ; when data is already sorted