

Algorithm for Quick Sort

QuickSort (A, p, r)

if $p < r$

Partition (A, p, r)

QuickSort ($A, p, q-1$)

QuickSort ($A, q+1, r$)

←→

Partition (A, p, r)

$x = A[r]$

$i = p-1$

for $j = p$ to $r-1$

$i = i+1$

Exchange $A[i]$ with $A[j]$

Exchange $A[i]$ with $A[r]$

Return $i+1$;

Algorithm of Merge Sort

Merge Sort (A, p, r)

if $p < r$

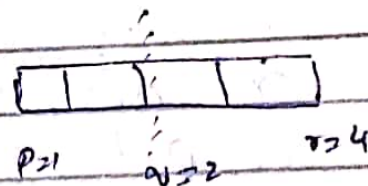
$$q = \frac{p+r}{2}$$

Merge Sort ($A, p, q-1$)

Merge Sort (A, q, r)

Merge (A, p, q, r)

←————→
Merge (A, p, q, r)



$$n_1 = q - p + 1$$

$$n_2 = r - q$$

Create new arrays $L[1, 2, \dots, n_1]$ $R[1, 2, \dots, n_2]$

for $i = 1$ to n_1

$$L[i] = A[p+i-1]$$

for $j = 1$ to n_2

$$R[j] = A[q+j]$$

$$L[n_1+1] = \infty$$

$$R[n_2+1] = \infty$$

$$i=1 \text{ \& \& } j=1$$

for $k = p$ to r

$$\text{if } L[i] \leq R[j]$$

$$A[k] = L[i]$$

$$i = i+1$$

else

$$A[k] = R[j]$$

$$j = j+1$$