Merge Sort and Merge

Merge Sort Algorithm

Algorithm 1 Merge-Sort(A[], p, r)
1: if p < r then
2:  \( q = \lfloor (p + r/2) \rfloor \)  \( \triangleright \) Divide .. Take floor
3:  Merge-Sort(A[], p, q) \( \triangleright \) conquer left half
4:  Merge-Sort(A[], q + 1, r) \( \triangleright \) conquer right half
5:  Merge(A[], p, q, r) \( \triangleright \) combine phase
6:  end if

Merge Algorithm

Algorithm 2 Merge(A[], p, q, r)
1:  \( n_1 = q - p + 1 \)
2:  \( n_2 = r - q \)
3:  let L[1..n_1 + 1] and R[1..n_2 + 1] be new arrays
4:  for i = 1 to n_1 do
5:      \( L[i] = A[p + i - 1] \)
6:  end for
7:  for j = 1 to n_2 do
8:      \( R[i] = A[q + j] \)
9:  end for
10: \( L[n_1 + 1] = \infty \)
11: \( R[n_2 + 1] = \infty \)
12: i = 1
13: j = 1
14: for k = p to r do
15:     if \( L[i] \leq R[j] \) then
16:        \( A[k] = L[i] \)
17:        i = i + 1
18:     else
19:        \( A[k] = R[j] \)
20:        j = j + 1
21:     end if
22: end for
Insertion Sort

Algorithm 3 Insertion-Sort(A[])

1: for j = 2 to n do
2:     key = A[j] \hspace{1cm} \triangleright \text{Insert } A[j] \text{ into the proper place in the array}
3:     i = j - 1
4:     while i > 0 and A[i] > key do
6:         i = i - 1
7:     end while
8:     A[i + 1] = key
9: end for

Selection Sort

Algorithm 4 Selection-Sort(A[])

1: for i = 1 to n - 1 do
2:     smallest = i
3:     for j = i + 1 to n do
4:         if A[j] \leq A[smallest] then
5:             smallest = j
6:         end if
7:     end for
8:     if smallest \neq i then
9:         hold = A[i]
11:        A[smallest] = hold
12:     end if
13: end for
Bubble Sort

Bubble Sort Algorithm

Algorithm 5 My-Bubble-Sort(A[])
1: \( n = A\.length \)
2: \( swapped = true \)
3: while ((swapped) and (n > 1)) do
4:    swapped = false
5:    i = 1
6:    while \( i < n \) do
7:        if \( A[i] > A[i + 1] \) then
8:            hold = \( A[i] \)
10:        \( A[i + 1] = hold \)
11:    swapped = true
12: end if
13: i = i + 1
14: end while
15: n = n − 1 \( \triangleright \) At this point the largest element is at the end of the partial array
16: end while