Data Structures and Algorithms
CMPSC 465

LECTURE 5, 9/2

Paul Medvedev

Merge Sort

```
MERGE(A, p, q, r)
 1   n₁ = q − p + 1
 2   n₂ = r − q
 3   let L[1 .. n₁ + 1] and R[1 .. n₂ + 1] be new arrays
 4   for i = 1 to n₁
 5       L[i] = A[p + i − 1]
 6   for j = 1 to n₂
 7       R[j] = A[q + j]          MERGE-SORT(A, p, r)
 8   L[n₁ + 1] = ∞                1   if p < r
 9   R[n₂ + 1] = ∞                2       q = ⌊(p + r)/2⌋
10   i = 1                        3       MERGE-SORT(A, p, q)
11   j = 1                        4       MERGE-SORT(A, q + 1, r)
12   for k = p to r               5       MERGE(A, p, q, r)
13       if L[i] ≤ R[j]
14           A[k] = L[i]
15           i = i + 1
16       else A[k] = R[j]
17           j = j + 1
```

The divide-and-conquer design paradigm

1. ***Divide*** the problem into subproblems that are smaller instances of the subproblem.
2. ***Conquer*** the subproblems by solving them recursively.
3. ***Combine*** subproblem solutions to give a solution to the original problem.

9/1/2015