Divide & Conquer Strategy

The divide-and conquer approach involves three steps at each level of the recursion.

(i) Divide the problem into a number of subproblems.

(ii) Conquer the subproblems by solving them recursively.

(iii) Combine the solutions to the subproblems into the solution for the original problem.

- The number of smaller instances into which the input is divided is k.
- For an input size n, let D(n) be the number of steps done by divide, and let C(n) be the number of steps done by combine.
- Then the general form of the recurrence equation that describes the amount of work done by the algorithm is

$$T(n) = D(n) + \sum_{i=1}^{k} T(\text{size}(I_i)) + C(n), \text{ for } n > \text{smallsize}$$

Divide & Conquer Skeleton

```
Solve (I)
    n = size (I);
    if (n <= smallsize)
        solution = directly solve (I);
    else
        divide I into I1, I2, .... Ik;
        for each i ∈ {1, 2, ... k}
            Si = solve (Ii);
        solution = combine (S1, S2, ... Sk);
    return solution;
```

A classic example of Divide & Conquer is Merge Sort. In Merge Sort, we divide array into two halves, sort the two halves recursively, & then merge the sorted halves.

eg: