Tutorial 7 (Analysis of Algorithms)
Thu. Apr. 20th, 2017

1. Consider the following recursive algorithm.

Algorithm 1 $Min1(A[0..n-1])$ ▷ Input: An array $A[0..n-1]$ of real numbers

1: **if** $n = 1$ **then**
2:      **return** $A[0]$
3: **else**
4:      $temp \leftarrow Min1(A[0..n-2])$
5:      **if** $temp \leq A[n-1]$ **then**
6:        **return** $temp$
7:      **else**
8:        **return** $A[n-1]$
9:      **end if**
10: **end if**

   (a) What does this algorithm compute?

   (b) Set up a recurrence relation for the algorithm's basic operation count and solve it.

2. Consider another algorithm for solving the same previous problem which recursively divides an array into two halves.

Algorithm 2 $Min2(A[l..r])$ ▷ Input: An array $A[0..n-1]$ of real numbers

1: **if** $l = r$ **then**
2:      **return** $A[l]$
3: **else**
4:      $temp1 \leftarrow Min2(A[l..\lfloor(l+r)/2\rfloor])$
5:      $temp2 \leftarrow Min2(A[\lfloor(l+r)/2\rfloor+1..r])$
6:      **if** $temp1 \leq temp2$ **then**
7:        **return** $temp1$
8:      **else**
9:        **return** $temp2$
10:      **end if**
11: **end if**