Quicksort Correctness Proof

Zezhou Jing

August 30, 2016

Theorem. *The quicksort algorithm correctly sorts every input array of length n, where $n \geq 1$.*

Proof. We approach the proof by mathematical induction.

The base case is that an array has only one element, namely $n = 1$. Every array with length 1 is trivially sorted. We can just return the input array without modifications. Thus the statement hold for $n = 1$.

We now wish to show the statement must hold for $n > k$ if it holds for $n = k$, where k is an arbitrary number bigger than 1. The quicksort algorithm partitions the input array around some pivot element p. Specifically, every element to the left of the pivot is smaller than the pivot while every element to the right of it is bigger. For convenience, let the lengths of the left and right parts be k_1, k_2, respectively. It is clear that k_1, $k_2 < n$.

By our inductive hypothesis, both two parts with lengths k_1 and k_2 are sorted correctly by recursive calls. Therefore the entire array of length $n = k_1 + k_2$ are sorted. That is, the statement holds for $n > k$ if it holds for $n = k$. Thus the quicksort algorithm is correct by principles of mathematical induction. $\square$

Remark. To clarify, suppose we have a pivot element p and two other elements l, r. l, the one to the left of the pivot is trivially sorted. So is the one to the right r. By quicksort, we have $l < p < r$. Since l and r are sorted, this (sub)array with three elements are sorted. Now proceed with induction.