Divide and Conquer

Spencer Compton

CPSI 2018: Squires Lecture 4

1 Introduction

Divide and Conquer is a technique common to many important algorithms, where we split harder problems into easier problems.

2 Merge Sort

A method of sorting commonly taught in beginning computer science classes is Merge Sort. At each iteration of merge sort, we split our current recurse on them. Our base-case is when we have an element by itself, and then we merge the two halves we recursed on with a two-pointer sweep.

3 Counting Inversions

Given an Array A, an inversion is classified as a i and j such that $i < j$ and $A[i] > A[j]$. To count inversions, we will consider inversions as arcs starting at the larger element and ending at the smaller element of each inversion. We will choose a split point in the middle of the array.

In $O(N)$, we will compute the number of arcs that start to the left of this split point and end to the right of this split point with a two pointer sweep of the sides when sorted. Then, we will recurse on both sides to find the arcs that (start on the left and end on the left) or (start on the right and end on the right). Since we do $O(N)$ work at each level, and split the intervals in halves every step, there is at most $O(\log N)$ height. Thus, $O(N \log N)$ work is done in total.

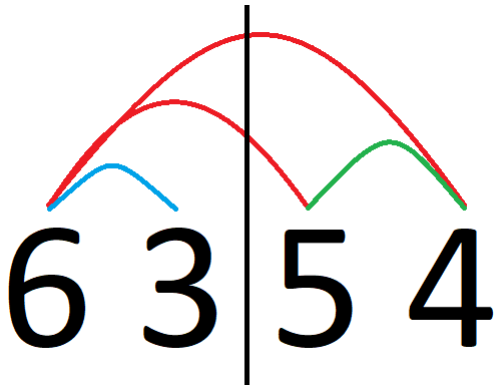

Figure 1: Visualization of inversions. Inversions found in the current step (starting on the left of the split point and ending on the right) are red. Inversions solely on the left are blue, and inversions solely on the right are green. The split point is represented as a vertical black line.

4 Centroid Decomposition

When we counted inversions on an array, we found all the arcs that crossed a split point and then split the array in half at that split point. We will use a similar technique to analyze paths with Centroid Decomposition. One question may be: it is easy to do for an array, how do we split a tree in half? The definition of a centroid is a node that, when remove, splits the tree into subtrees of size at most $\frac{N}{2}$. This means if we remove the centroid, and then remove the centroids of those subtrees, and so on, a node will only ever have been in at most $O(\log N)$ subtrees. Just as we only found arcs that went over split points, we will analyze paths that go through the centroid, remove the centroid, and recurse.

Consider the following slight modification to IOI 2011 Race:

- Given a tree of size $N <= 2 \times 10^5$, with $N - 1$ weighted edges, what is the shortest (in terms of number of edges) path whose edge-weights sum exactly to K.

5 Common Techniques

- Divide and conquer on ranges.
- Divide and conquer on answers.
- Analyzing paths with centroid decomposition.

6 Practice Problems

USACO Platinum 2017 February: Why Did the Cow Cross the Road

IOI 2011: Race

USACO Platinum 2016 February: Circular Barns