1 Find Rice Subarrays

To solve this problem, I just needed to use logic and implement scanning from left side to right side to check every possible subarray. But it was slow, so I optimized by neglecting cases where the middle element is bigger than the left boundary. Also, another way was to solve through stacks, but this code already worked and its simple and understandable. So the early exit method was really efficient.

2 Split Sort

Firstly, thinking about all methods, I tested this first. I kept track of groups of numbers and splits them whenever two positions u and v were inside of the group. Then I find the range between u and v and cut that group into two smaller sets (the middle part and the rest), and count the energy. Actually for storing each group, I used Hashmap. Then I tried other method by containing all elements into one group. If both elements belong to the same group, it finds the range between them and splits that group into two sets. I optimized it by always reusing the larger group's ID and only assigning a new ID to the smaller one which is efficient for memory. Also I tried using with dictionaries, sets, lists and etc. So my final code is same logic but with optimizing through fast reading in Python and improved efficiency by comparing group size with the split range size to decide which way to loop which reduces unnecessary checks. Other possible ways to solve it through Binary Tree, but professor kinda scared us. Also Linked List method is good. But mine even if it's a bit slow, memory is used little, simple and understandable and doesnt take too much lines to code.