This part I will calculate the running time of this algorithm. Note that m is the size of array A needed to be grouped, and n is the number of groups.

First, the algorithm will use “if” sentence to find the proper case. The running time of this is . The algorithm will consider three cases. I find that the latter two cases are not common cases and their running time is not large, so next step I will mainly consider the first case: if .

Then, I find that the first “for” loop in the first case needs to sum up all the numbers in the array A and it will repeat m times, so in this operation the running time is .

Next, there are two “for” loops between which one (from 1 to m-1) is nested in another (from 1 to n-1). And the operation needs to find the maximum number from an array whose range is from j-1 to i. And the worst case we need to sum up the array’s numbers whose range are from 0 to m-1(can be nearly the total array). So this part the running time is .

At last, the “for” loop is from n-2 to 0, so the running time is .

Totally the sum of four sections’ running time can be . If n is smaller enough than m, the running time can be . But n is a randomized number, so ultimately the running time of the algorithm is .