

All the Trades Are the Best

Input file: standard input
Output file: standard output
Time limit: 4 seconds
Memory limit: 512 megabytes

You are a trader at an investment company. One day, you discover that the stock you're responsible for trading follows an exactly repeating daily pattern: the day is divided into n equal time intervals, and the stock's price during the i -th interval is always a_i . As the n -th interval of one day concludes, the first interval of the next day immediately follows, with the price shifting from a_n to a_1 . This is a once-in-a-lifetime opportunity — if you can just buy and sell according to this pattern you've already figured out, during your working hours, there's a guaranteed way to profit!

Your working hours vary, and a working period may extend past midnight. You are given q working periods. The i -th period starts at slot l_i and ends at slot r_i , including both endpoints. The slots in chronological order are:

- $l_i, l_i + 1, \dots, r_i$ if $l_i \leq r_i$;
- $l_i, l_i + 1, \dots, n, 1, 2, \dots, r_i$ if $l_i > r_i$.

In each working period, you may complete at most k transactions. Each transaction consists of buying one share and selling it in a later time slot, with profit equal to the sale price minus the purchase price. You may hold at most one share at a time, so you must sell your current share before buying another. All purchases and sales must take place within that working period and follow the chronological order above. You may also make no transactions, earning a profit of 0.

For each working period, independently find the maximum total profit you can earn.

Input

The first line contains three integers n, k, q ($1 \leq n \leq 10^5$, $1 \leq k \leq 800$, $1 \leq q \leq 3 \times 10^5$): the number of time slots in a day, the maximum number of transactions per working period, and the number of working periods, respectively.

The second line contains n integers $a_1, a_2, \dots, a_n$ ($1 \leq a_i \leq 10^6$), where a_i is the stock price in the i -th time slot of each day.

Each of the next q lines contains two integers l_i, r_i ($1 \leq l_i, r_i \leq n$), describing the i -th working period.

Let len_i be the number of time slots in this period. If $l_i \leq r_i$, then $\text{len}_i = r_i - l_i + 1$; otherwise, $\text{len}_i = n - l_i + 1 + r_i$.

For each working period, l_i and len_i are generated independently and uniformly at random from $[1, n]$ and $[\max(1, \lfloor 0.15n \rfloor), \max(1, \lfloor 0.85n \rfloor)]$, respectively. The value of r_i is uniquely determined by l_i and len_i . Specifically, when the interval is $[1, 1]$, the random result will always be 1.

There are exactly 50 test cases, excluding the sample.

Output

Output q lines. The i -th line should contain one integer: the maximum profit you can earn during the i -th working period with at most k transactions.

Example

standard input	standard output
6 2 3	7
3 1 4 1 5 9	4
2 5	0
5 2	
4 4	