Interval Query (Sqrt Decomposition Ver.)

Time Limit: 1 Second
Memory Limit: 256 MB

You are asked to solve a classic problem: given an array $a_1, \ldots, a_n$ of n elements and m intervals $(l_1, r_1), \ldots, (l_m, r_m)$ $(\forall i \in \{1, \ldots, m\}, l_i \le r_i)$. For each interval (l_i, r_i), find the sum of element of $a_{l_i}, \ldots, a_{r_i}$.

To make sure you are really using sqrt decomposition to solve this problem, instead of finding $\sum_{i=l}^{r} a_i$, we ask you to find $\sum_{i=l}^{r} (-1)^{\mathrm{lowbit}(l \oplus \lfloor \frac{i}{\sqrt{n}} \rfloor)} a_i$, where $\oplus$ is bitwise XOR, and $\mathrm{lowbit}(x) = x \ \& \ -x$ gets the lowest set bit in a binary number.

Input

The first line contains two integers n and m ($1 \le n, m \le 5 \times 10^4$, $\sqrt{n}$ is an integer) - the number of elements in the array and the number of intervals.

The second line contains n integers $a_1, \ldots a_n$ ($1 \le a_i \le 10^9$) - the elements in the array.

The last m lines describe the intervals. The i-th line contains two integers l_i and r_i ($1 \le l_i \le r_i \le n$), denoting the boundary of the i-th interval.

Output

For each interval, output a single integer denoting the sum $\sum_{i=l}^{r} (-1)^{\mathrm{lowbit}(l \oplus \lfloor \frac{i}{\sqrt{n}} \rfloor)} a_i$.

Sample Inputs

```
4 3
3 2 1 3
1 3
2 3
2 4
```

Sample Outputs

```
0
-3
0
```