E Crack Free Square Cover

Time limit: 1.0s
Memory limit: 256MB

We have an infinite 2D wall with n horizontal and m vertical infinite lines. We want to fill the maximum area of this wall with some (possibly zero) number of squares such that the following criteria hold:

1. The squares should not cover the same area twice.
2. Squares should not intersect with the lines (the lines can be touching the squares).
3. Each of the squares should have at least one of the $n \times m$ line intersections as one of its corners.
4. Squares should be inside the rectangle $((a, c), (b, d))$, the rectangle where its down-left corner is the point (a, c) and its up-right corner is the point (b, d). **Read the inputs section carefully.**

What is the maximum achievable sum of the area of the squares?

Input

The first line contains two integers n and m $(2 \leq n, m \leq 5 \times 10^5)$ — the number of vertical and horizontal lines respectively.

The second line contains n integers $y_1, y_2, \ldots, y_n$ $(0 \leq a = y_1 < y_2 < \ldots < y_n = b \leq 10^6)$ meaning the i-th horizontal line is $y = y_i$.

The third line contains m integers $x_1, x_2, \ldots, x_m$ $(0 \leq c = x_1 < x_2 < \ldots < x_m = d \leq 10^6)$ meaning the i-th vertical line is $x = x_i$.

It is guarantied that x_i-s and y_i-s are even.

Output

Print a single integer, the maximum achievable sum of areas of the squares.

It can be proven that with the given constraints the answer is always an integer.

Samples

Sample input 1	Sample output 1
4 7 0 4 6 10 0 2 4 6 10 12 18	176

Sample input 2	Sample output 2
3 3 0 300000 1000000 0 400000 1000000	930000000000

Sample input 3	Sample output 3
5 5 0 41776 90818 98832 100000 0 5836 35546 50738 100000	8287729808