

Children in a kindergarten have received a large sack containing **M** candies. It has been decided that the candies are to be distributed among **N** children.

Each child has stated the number of candies that it wants. If a child isn't given the amount of candy it wants, it will get angry. In fact it'll get angrier for each candy it is deprived of. Some speculate that it's anger will be equal to the square of the number of candy it is deprived of. For instance, if Mirko states that he wants 32 candies but receives only 29, he would be missing 3 candies, so his anger would be equal to 9.

Unfortunately, there is an insufficient amount of candy to satisfy all children. Therefore, the candies should be distributed in such a way that the sum of the children's anger is minimal.

INPUT

The first line contains two integers, **M** ($1 \leq M \leq 2 \cdot 10^9$) and **N** ($1 \leq N \leq 100\,000$).

The following **N** lines contain integers (one per line) which represent the wishes of the children. Those numbers are all strictly less than $2 \cdot 10^9$, and their sum always exceeds **M**.

OUTPUT

The first and only line of output must contain the minimum sum of the children's anger.

Note: The test cases will ensure that the result fits in a 64-bit unsigned integer: *int64* in Pascal, *long long* in C/C++, *long* in Java.

SCORING

Test cases worth 40% of total points have **M** not greater than 200 000.

Test cases worth 70% of total points have no child state that it wants more than 100 000 candies.

Test cases worth 80% of total points have at least one of the above stated constraints will be met.

SAMPLE TESTS

input	input
5 3	10 4
1	4
3	5
2	2
	3
output	output
1	4