Problem E. The Profiteer

Input file:	*standard input*
Output file:	*standard output*
Time limit:	2 seconds
Memory limit:	512 mebibytes

BaoBao has a store. There are n items in the store, labeled by $1, 2, \ldots, n$. The value of the i-th item is v_i, and the price of it is a_i dollars. JB is planning to visit BaoBao's store tomorrow. JB always buys items optimally. Assume JB has t dollars, he will buy a set of items such that the total value is maximized and the total price is no more than t.

The profiteers cheated people right and left. BaoBao knows JB is rich, so he decides to choose a pair of integers l and r, where $1 \le l \le r \le n$, and raises the prices of all the items indexed in $[l, r]$. When JB comes tomorrow, he will need to pay b_i dollars instead of a_i dollars for the i-th item, where $l \le i \le r$.

However, BaoBao doesn't know how rich JB is, he only knows t is an integer uniform randomly chosen in $[1, k]$. BaoBao doesn't want JB to buy so many good items, he is now wondering how many pairs of integers l and r he can choose such that the expected total value of JB's shopping list $\frac{f(1)+f(2)+\cdots+f(k)}{k}$ will not exceed E, where $f(t)$ denotes the total value of the shopping list when JB has t dollars. Please write a program to help BaoBao.

Input

The input contains only a single case.

The first line contains three integers n, k and E ($1 \le n, k \le 200\,000$, $n \times k \le 10^7$, $1 \le E \le 10^9$).

Each of the following n lines contains three integers v_i, a_i and b_i ($1 \le v_i \le 10\,000$, $1 \le a_i < b_i \le k$), denoting the value, the initial price and the raised price of the i-th item.

Output

Print a single line containing an integer, denoting the number of valid pairs of integers l and r.

Examples

standard input	standard output
4 5 3 3 2 4 1 2 3 2 1 2 3 1 3	1
4 5 4 3 2 4 1 2 3 2 1 2 3 1 3	3