Chirag's final assignment (Hard)

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	256 megabytes

Note: This is the Hard version. The only difference between the easy and hard versions is the constraints.

Chirag is almost done with college and has one final assignment left. In the assignment, he has to traverse a weighted undirected graph. However, Chirag did not know how to traverse a weighted graph, so he simplified the problem to an undirected unweighted graph such that there are $w - 1$ nodes between two nodes if they had an edge of weight w. The problem he needs to solve now is how many nodes will be visited if he starts BFS from node 0 and stops at layer k, i.e., the shortest path between any visited node and 0 does not exceed k.

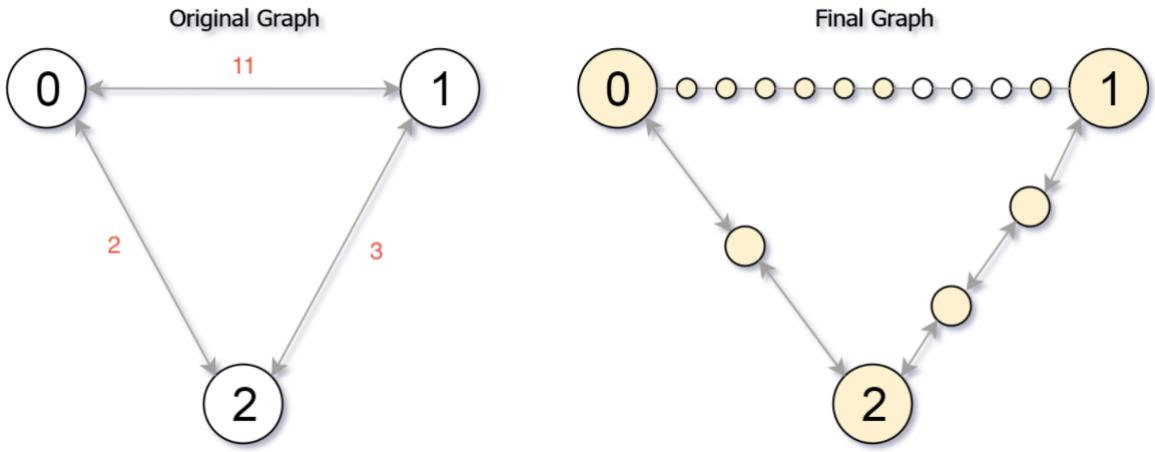
Input

The first line contains n, e and k, where $2 \leq n \leq 1000$ is the number of nodes in the original graph (not the graph modified by Chirag), $1 \leq e \leq n * (n-1)/2$ is the number of edges in the original graph and $1 \leq k \leq 10^9$ is the layer at which Chirag stops the bfs.

The next e lines contain v_i, v_j and w_{ij} denoting $2 \leq w_i \leq 10^4$ denoting integer edge weight.

Output

Print a single integer *visited* where 1 <= *visited* <= 10^{12} is the maximum number of nodes the bfs would visit.



Examples

standard input	standard output
3 3 6 0 1 11 0 2 2 1 2 3	13
4 4 10 0 1 5 1 2 7 0 2 9 1 3 2	23

Note

The number of visited nodes includes the nodes Chirag has added.