Problem H
Railway Tickets

Input File: H.in
Output File: standard output
Time Limit: 0.5 seconds (C/C++)
Memory Limit: 256 megabytes

The common rule for transporting passengers in long-distance trains at many railways is that each ticket should specify a reserved place. It's rather convenient for passengers to know that they will have a guarantee for a vacant place beforehand. However, such a rule can pose the problem of false lack of places.

Consider such an unlikely but still possible case. A train with two seats goes from A to C, with only one intermediate stop at station B. Suppose seat 1 is bought for the A-B segment, and seat 2 is bought for the B-C segment. Therefore, there is no available seat between A and C. However, a traveler can buy seat 2 for the A-B segment and seat 1 for the B-C segment, and move from one place to another during the trip.

Your task is to write a program, which, knowing which tickets are already sold, finds the number of source-destination pairs of stations which can still be reached by buying two or more tickets and switching places. You may assume that any seat which is not sold between a specific source-destination may be bought.

Input
The 1st input line contains the number **K** ($3 \le \mathbf{K} \le 1000$) of places in the train; the 2nd line contains the number **N** ($3 \le \mathbf{N} \le 10000$) of stations in the train's route; the 3rd line contains the number of the tickets already sold **T** ($0 \le \mathbf{T} \le \min(10^5, \mathbf{K}(\mathbf{N}-1))$). Each of the next **T** lines contains three integer numbers **pl**, **st**, **fn**, describing tickets: **pl** stands for the number of the reserved place (assuming that places are sequentially numbered in range 1 to **K** over all the train, without dividing by carriages), and **st, fn** are the departure, destination stations, respectively (stations are numbered along the train's route consequently from 1 to **N**). In each ticket, **st** < **fn**, i. e. arrival station's index is strictly greater than departure station's one. Different tickets for the same place are possible only if their travel ranges do not overlap (each next ticket can start either at the station where the previous ticket for the place ends, or somewhere later on the route).

Output
Print one integer — the requested number of stations pairs.

Sample input	Sample output
3 10 6 2 9 10 3 5 9 1 2 10 2 1 4 2 7 8 3 1 2	10

Note These 10 pairs are: (1, 3), (1, 4), (1, 5), (1, 6), (1, 7), (2, 6), (2, 7), (3, 6), (3, 7), and (8, 10). For other pairs of stations, it's either possible to buy a direct ticket with specific reserved place or otherwise the train has no vacant places, even for a passenger, who is ready to buy separate tickets and change his or her place during the trip.