Today is the long-awaited day for all school students – the first holiday for the new school year. Our main heroine **D**eni is now in grade 10. She prepared for today – she found that there are ***N*** stores downtown and she plans to visit some of them with her friends. But **D**eni and her friends don't like some of the connections between the stores and they won't use them. So they have made a list of ***M*** pairs of stores such that a pair (***x***, ***y***) is from the list if they like the connection from store ***x*** to ***y*** and of course they can reach store ***x*** from **y** and for every pair they have determined the time for travelling the connection (it is the same in both directions). There are no pairs with stores with the same numbers and there are no duplicate pairs.

Deni is very superstitious and one of the superstitions in which she believes is that the total time for travelling must be divisible by ***D***. **D**eni and her friends don't have unlimited time so the maximum time they can spend travelling is ***K***. As all girls, **D**eni is very curious and she starts to count the number of different routes for visiting some of the stores (a store can be visitted more than once). Unfortunately, this number can be large so **D**eni remembers that she knows you – very good programmer and asks you to write the program **superstition,** which counts the number of valid routes. One route is valid if it uses the connections from the list of pairs, the total time for travelling is divisible by ***D*** and it is ≤ ***K***. Two routes are different if there is a difference in the sequences of visited stores. You immediately notice that the answer can be very large and thus **D**eni tells you that she only wants the remainder when the answer is divided by 1,000,000,007.

Input

From the first line of the standard input read four integers ***N***, ***M***, ***D*** and ***K***. From each of the next ***M*** lines read three integers x_i, y_i and t_i – bidirectional connection between x_i and y_i with time of travel t_i ($1 \le i \le M$).

Output

The number of different valid routes. Since this number may be quite large, you are required to print its remainder when divided by 1,000,000,007.

Constraints

- $2 \le N \le 80$
- $2 \le M \le 3160$
- $2 \le D \le K \le 10^9$
- $1 \le t_i \le 10$

Subtasks and grading

Subtask	Points	*N*	*M*	*D*	*K*	Further constraints
1	5	≤ 5	≤ 10	≤ 12	≤ 12	There are no further constraints.
2	30	≤ 80	≤ 3160	$\le 10^4$	$\le 10^4$	There are no further constraints.
3	10	≤ 20	≤ 190	$\le 10^9$	$\le 10^9$	***D*** = ***K*** and $\sum_{i=1}^{M} t_i \le 200$.
4	20	≤ 20	≤ 190	$\le 10^9$	$\le 10^9$	$\sum_{i=1}^{M} t_i \le 200$.
5	15	≤ 30	≤ 435	$\le 10^9$	$\le 10^9$	***D*** = ***K***.
6	20	≤ 30	≤ 435	$\le 10^9$	$\le 10^9$	There are no further constraints.

Your program will get points for a given subtask only if all test cases for that subtask are passed successfully.

Examples

Input	Output	Explanation of the example
3 3 2 2 1 2 1 2 3 2 3 1 1	8	Here ***D*** = ***K*** = 2 i.e. the needed routes are only with total time 2. They are: 1 – 2 – 1 2 – 1 – 2 3 – 1 – 3 1 – 3 – 1 2 – 3 3 – 2 2 – 1 – 3 3 – 1 – 2 Notice that stores and connections can be repeated more than once.
5 7 5 10 1 3 8 2 5 7 3 4 3 1 4 2 2 3 1 1 5 4 4 5 4	58	Because ***D*** < ***K*** the needed routes are with total time 5 and 10.
5 9 2 20 1 2 1 2 3 2 3 1 1 3 4 1 4 5 2 5 3 1 1 5 1 2 4 1 2 5 1	989802661	Here the real answer is a big number so the output is only its remainder when divided by 1,000,000,007.
5 7 5000000 5000000 1 3 8 2 5 7 3 4 3 1 4 2 2 3 1 1 5 4 4 5 4	598634781	Here the real answer is a big number so the output is only its remainder when divided by 1,000,000,007.