Problem J. Steiner Tree in Random Graph

Input file:	`random-steiner.in`
Output file:	`random-steiner.out`
Time limit:	2 seconds *(3 seconds for Java)*
Memory limit:	256 mebibytes

You are given a weighted graph on n vertices. Your task is to find a connected subgraph of minimal total weight which contains all vertices with numbers from 1 to $n - k$ inclusive.

To make the problem easier to solve, and the tests easier to generate, all tests except the examples are generated using the following algorithm:

- Integers n, m, and k are chosen by the jury.
- All $\frac{n \cdot (n-1)}{2}$ possible edges are considered in random order.
- When being considered edge is added to the graph if, after it is added, it is still possible to add zero or more of the following edges and get a connected graph with at most m edges in total.
- For each edge, its weight is an integer chosen uniformly at random from the segment $[50, 100]$, independently from other edges.

Input

The first line contains three integers n ($5 \le n \le 100$), m ($2 \cdot n \le m \le 10 \cdot n$) and k ($0 \le k \le 30$). Each of the next m lines holds three integers a_i, b_i and w_i ($1 \le a_i < b_i \le n$, $50 \le w_i \le 100$) which mean that vertices a_i and b_i are connected by an edge of weight w_i.

It is guaranteed that the graph has no loops and no multiple edges. Additionally, it is guaranteed that in all tests except the examples, the graph is generated by the algorithm described above.

Output

On the first line, print one integer c, the number of edges in the chosen subgraph. On the next c lines, print the chosen edges: the numbers of vertices connected by each of them. If there are several possible answers, print any one of them.

Examples

random-steiner.in	random-steiner.out
5 10 3 1 2 50 1 3 51 1 4 52 1 5 53 2 3 54 2 4 55 2 5 56 3 4 57 3 5 58 4 5 59	1 1 2
5 10 2 1 2 100 1 3 100 1 4 57 1 5 56 2 3 100 2 4 54 2 5 53 3 4 52 3 5 51 4 5 50	3 5 2 5 3 1 5