Yet another computer network problem

ACRush and Jelly are practising in the computer room for next TCO. Suddenly, they found the network is very slow. After a few diagnoses, they realized that there are many redundant wires. So they plan to repair the network, change it to an optimal tree topology. And they can't spend too much money to purchase wires. Then.. too easy? Are you thinking about minimum spanning tree?

But the real trouble is the connectors have their own limitation. They can only allow one computer connects with at most *B* computers.

There are totally 10 cases, arranged in increasing order of the size of *N* (number of computers). Weight of case i-th is w[i] = i. We define *infinity* = 4 * 10^9. And in a tree, let's call number of computers that computer i connects with is *degree* of computer i.
For case i-th you must show us a satisfied tree with total cost C[i] and maximum degree M[i], considering all computers of that tree.
The formula to compute score is as below:
With case i-th:
If your M[i] <= B then Score[i] = w[i] * C[i]
If your M[i] > B then Score[i] = (w[i] + 10) * C[i] * M[i]

To make the challenge more interesting, with a simple brute force program, we generated 10 upper bound degrees U[i] (1 <= i <= 10) for each of 10 cases.
For any case i-th:
If your M[i] > U[i] then Score[i] = infinity
Finally, TotalScore = (Score[1] + Score[2] + ... + Score[10]) / 10
Try to minimize the *TotalScore*.

Input

First line contains 3 integers *N, M, B* -- number of computers, number of pairs of computers can be connected and the maximum number of computers that a computer can connect with. (1 <= *N* <= 10^4, 1 <= *M* <= 10^5, 1 <= *B* <= N)
Next *M* lines, line i-th contains a triple (u[i], v[i], c[i]) -- means if we want to connect computers u[i] and v[i] we should purchase a wire, cost c[i] (1 <= u[i], v[i] <= N, 1 <= c[i] <= 20000). The wires are bidirectional.

Output

The first line contains 2 numbers --- total cost of your tree and the maximum degree in all computers of that tree. Next, print *N*-1 lines, corresponding to *N*-1 edges of the tree, each edge on one line, forms *u v*.

Example

Input:
3 3 2
1 2 1
2 3 1

1 3 5

Output:
2 2
1 2
2 3