Problem : MST algorithms Implementation

Problem Description

Given a connected and weighted graph, you are to make a program which finds a minimum spanning tree (MST) using both Kruskal and Prim algorithms.

Input

The name of input file is 'mst.inp.' The input file has the information of a graph $G=(V,E)$. In the first line of the input, two integers n and m are given, where $n=|V|$ and $m=|E|$. Note the indices of vertices start with 0, which means $V=\{v_{0,}\cdots,v_{n-1}\}$.

In the following m lines, each of which contains the information of a weighted edge. Each weighted edge consists three integers, u, v, and w. A triple (u,v,w) denotes that there is an edge of weight w between vertices u and v.

Even though the edge numbers are not specified, each edge in input is assigned with unique edge number. The edge number of the first input edge is considered 0, the next 1, and so on.

Note that $1 \le n \le 10{,}000$, $1 \le m \le 100{,}000$ and $1 \le w \le 10{,}000$.

Output

The name of output file is 'mst.out.'

First, show the result obtained by Kruskal algorithm. In the first line, print "`Tree edges by Kruskal algorithm: `" followed by the sum of edge weights in the spanning tree you have selected as shown in the following sample. In the following $n-1$ lines, show the edge numbers, each in a line, according to the order you selected for the minimum spanning tree. If there are two or more edges with the same weight to be selected at k^{th} step while you are applying Kruskal algorithm, select the edge with the smallest edge number. Remember the assumption that the edge number of the first input edge is 0, the next 1, and so on.

Then, show the results obtained by applying Prim algorithm three times with different starting vertices. First, Apply Prim algorithm with the starting vertex v_0. In the first line, print "`Tree edges by Prim algorithm with starting vertex 0: `" followed by the sum of edge weights in the spanning tree you have selected, where 0 means the index of the starting vertex. In the following $n-1$ lines, show the edge numbers, each in a line, according to the order you selected for the minimum spanning tree. If there are two or more edges with the same weight to be selected at k^{th} step while you are applying Prim algorithm, select the edge with the smallest edge number. Don't forget the assumption that the edge number of the first input edge is 0, the next 1, and so on.

Then, apply Prim algorithm with the starting vertex $v_{n/2}$. In the first line, print "`Tree edges by Prim algorithm with starting vertex k: `", where k is $\frac{n}{2}$, the index of the starting vertex, followed by the sum of edge weights in the spanning tree you have selected.

Similarly, apply Prim algorithm with the starting vertex v_{n-1} and show the results similarly.

The following shows sample input and output.

Sample Input	Output for the Sample Input
7 21 4 1 15 4 0 19 3 1 18 6 1 12 6 5 13 2 6 10 3 0 18 3 6 17 3 2 16 4 3 16 0 6 19 5 1 12 0 1 13 2 0 19 1 2 10 4 2 17 2 5 13 5 4 14 5 3 17 0 5 10 4 6 17	Tree edges by Kruskal algorithm: 72 5 14 19 11 17 8 Tree edges by Prim algorithm with starting vertex 0: 72 19 11 14 5 17 8 Tree edges by Prim algorithm with starting vertex 3: 72 8 5 14 11 19 17 Tree edges by Prim algorithm with starting vertex 6: 72 5 14 11 19 17 8

Constraints: The file name should be 'mst.{c, cpp, java}.'