#include<bits/stdc++.h>

using namespace std;

vector<vector<pair<int,int> > > graph;

void prims\_algo(int src, vector<int> &key, vector<bool> &mst, vector<int> &parent){

for(int i=0;i<graph.size()-1;i++){

int mini = INT\_MAX,u;

for(int j=0;j<key.size();j++){

if(mst[j]==false && key[j]<mini){

mini = key[j];

u = j;

}

}

mst[u] = true;

for(auto x: graph[u]){

if(mst[x.first]==false && key[x.first]>x.second){

parent[x.first] = u;

key[x.first] = x.second;

}

}

}

}

int main(){

int n;

int e;

cin>>n>>e;

graph.resize(n);

for(int i=0;i<e;i++){

int x,y,w;

cin>>x>>y>>w;

graph[x].push\_back(make\_pair(y,w));

graph[y].push\_back(make\_pair(x,w));

}

int src =0;

vector<int> key(n,INT\_MAX);

vector<int> parent(n,-1);

vector<bool> mst(n,false);

key[src] = 0;

mst[src] = true;

prims\_algo(src,key,mst,parent);

for(int i=1;i<n;i++){

cout<<parent[i]<<" "<<i<<endl;

}

return 0;

}