//Minimum Spanning Tree

//NOTE: IMPLEMENT UNION DISJOINT FUNCTIONS FIRST

#define MAX 1000

vector<pair<int, pair<int, int> > > Edge;

int main()

{

int V, E, u, v, w;

scanf("%d %d", &V, &E);

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

scanf("%d %d %d", &u, &v, &w);

Edge.push\_back(make\_pair(w, make\_pair(u, v)));

}

sort(Edge.begin(), Edge.end()); //sort according to weight min to max

int mst\_cost = 0, selected\_edge = 0;

unionInit(V);

for(int i = 0; i < E && selected\_edge < V; i++) {

u = Edge[i].second.first;

v = Edge[i].second.second;

w = Edge[i].first;

if(!isSameSet(u, v)) {

selected\_edge++;

mst\_cost += w;

makeUnion(u, v);

}

}

printf("MST in Kruskal : %d\n", mst\_cost);

Edge.clear();

return 0;

}