朴素Prim

```cpp
#include <bits/stdc++.h>
using namespace std;

const int N = 505, INF = 0x3f3f3f3f;
int g[N][N];
int dist[N];
bool vis[N];
int n, m;

int prim(){
    memset(dist, 0x3f, sizeof dist);
    int ans = 0;
    for(int i = 0; i < n; i++){
        int t = -1;
        for(int j = 1; j <= n; j++){
            if(!vis[j] && (t == -1 || dist[j] < dist[t])){
                t = j;
            }
        }
        if(i && dist[t] == INF) return INF;
        if(i) ans += dist[t];

        for(int j = 1; j <= n; j++){
            dist[j] = min(dist[j], g[t][j]);
        }

        vis[t] = true;
    }
    return ans;
}

int main(){
    cin >> n >> m;
    memset(g, 0x3f, sizeof(g));
    for(int i = 0; i < m; i++){
        int u, v, w;
        scanf("%d%d%d", &u, &v, &w);
        g[u][v] = min(g[u][v], w);
        g[v][u] = g[u][v];
    }

    int t = prim();

    if(t == INF) printf("orz");
    else cout << t;
    return 0;
}
```