Greedy techniques

Prim's algorithm

Q2.
N - 1
Q5.
Click submit
Q7.

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

const int N = 2e5 + 10;
vector<bool> vis(N, false);
vector<pair<int,int>> adj[N];

long long prims(int source, int n) {
    priority_queue<pair<int, int>, vector<pair<int, int>>, greater<pair<int, int>>> pq;
    pq.push({0, source}); // {weight, vertex}
    long long mstWeight = 0;

    while (!pq.empty()) {
        int u = pq.top().second;
        int w = pq.top().first;
        pq.pop();

        // Skip if the vertex is already visited
        if (vis[u]) continue;
        vis[u] = true;
        mstWeight += w;

        // Traverse all adjacent vertices of u
        for (auto edge : adj[u]) {
            int v = edge.first;
            int weight = edge.second;
            if (!vis[v]) {
                pq.push({weight, v});
            }
        }
    }

    return mstWeight;
}

int main() {
    int n, m;
    cin >> n >> m;
    for(int i = 0; i < m; i++) {
        int x, y, w;
        cin >> x >> y >> w;
        adj[x].push_back({y, w});
        adj[y].push_back({x, w});
    }

    long long mstWeight = prims(1, n);
    cout << mstWeight << endl;

    return 0;
}
```

Q8.(**Change the language to python)**
https://codefile.io/f/VEFxJdrXII

Kruskal's Algorithm

Q3.
Click submit
Change the language to python
Q5.
https://codefile.io/f/bxKIEEqXXQ
Q6.
https://codefile.io/f/QR67CmnCtt

Huffman Trees and codes

Q2.
Both First and Second
Q6.
Click submit